Obelisk
Project Final Report

Table of Contents

Acronyms and Terms

HotSync   A method of exchanging data between a handheld and a Computer.

I/O   Input/Output

IrDA   Infrared Data Association

IrTran-P   Infrared Transfer Protocol

JPEG   Joint Picture Expert Group

MSE   Master of Software Engineering

OS   Operating System

PAMD   Plug-in Architecture for Mobile Devices

PC   Personal Computer

PDA   Personal Digital Assistant

RFP   Request For Proposal

SEI   Software Engineering Institute

UI   User Interface

Win 2K   Windows 2000

Win 9X   Windows 95 and Windows 98

Win ME   Windows Millennium Edition

Win XP   Windows XP

1   Introduction

The purpose of this document is to present the various requirements and constraints that need to considered while developing architecture for the proposed software. This document will not describe the implementation details unless they directly affect the architecture.

1.2   Project Description

The software project itself deals with writing companion applications for the Casio WQV-3 and WQV-10 camera watches. The first companion application is a Palm application that allows a user to transfer images from the watch to a PDA and then perform editing, viewing, and other operations. The second application provides a way for images to be transferred between the PDA and a PC. The following scenario provides a good overview of the requirements for the two applications.
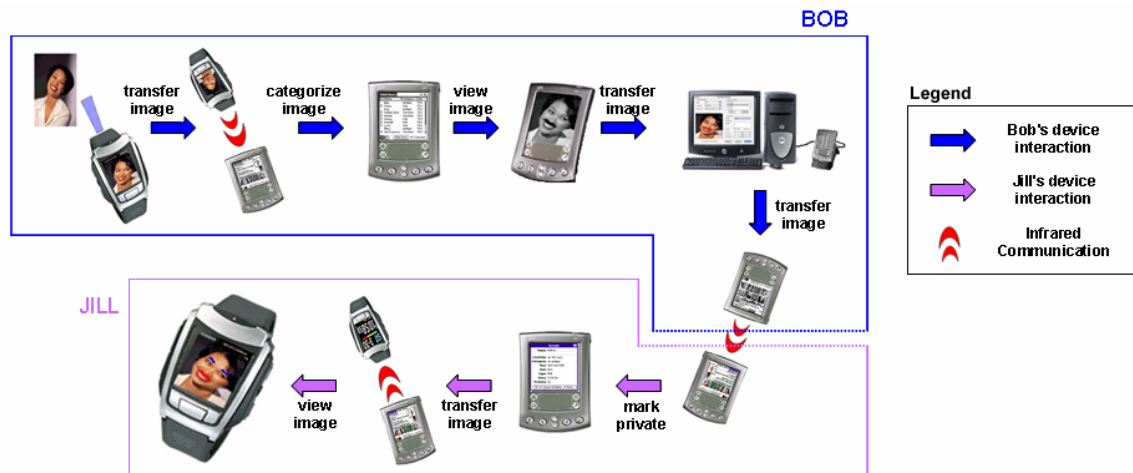
Figure 1: Project scenario

The scenario in figure 1 contains the following steps:
1. Bob takes a picture of Jill using the Casio watch
2. Bob transfers the image to his PDA
3. Bob categorizes the image in his "loved ones" category
4. Bob views the image on his PDA (image is shown in black and white as a result of his PDA hardware limitations)
5. Bob adds a red mustache and blue eyebrows and saves the image
6. Bob HotSyncs his PDA with his PC (image is transferred to his PC)
7. Bob directs his PDA toward Jill's PDA and transfers the modified image to Jill
8. Jill is marks the image as private
9. Jill transfers the image to her Casio watch
10. Jill views the image in full color on her watch

1.3 Context

The project can be seen from two different perspectives. One is the real perspective, where the clients are researchers from the SEI and are interested in producing a technical report about the way decisions are made by software development teams. The other is a virtual face in which the Casio Corporation is the client and is interested in the quick release of the companion software to facilitate an early release of the camera watch.

The business context of the project is best understood using the virtual face. The Casio Corporation is hoping that offering the companion software with their camera watches will increase sales of the watches. These watches will be of greater use if users are able to do more things with the images taken. It is essential to deliver the companion applications, WQV-Buddy, before August 2003, which is the advertised watch release date. The watch documentation and advertisements already include the WQV-Buddy software.

2 Requirements

The following lists of requirements are the prioritized high-level requirements for the Palm and PC applications. The requirements are prioritized in descending order of importance. For a more detailed list of requirements see Appendix A.

2.1 Palm Application

This section lists the functional requirements, architectural requirements, and constraints for the Palm application.

2.1.1 Functional Requirements

R1. Transfer images from the watch
R2. Transfer images to the watch
R3. View images in list view
R4. Edit image graphically in 24-bit color
R5. Beam a single image or a category of images to another PDA (Interoperability)
R6. Control the watch remotely (Interoperability)
R7. View images in thumbnail view
R8. Change an image's security setting using the Palm OS security feature
R9. Provide the ability to undo graphical edits
R10. Delete a single image or a category of images
R11. Transfer images to a PC over IrTran-P (Interoperability)
R12. Sort images
R13. View images in slideshow view
R14. Edit image metadata
R15. Create a new image using a specified background
R16. Change an image's category
R17. Edit, create, and delete categories

2.1.2 Architectural Requirements

A1. Compress/decompress images in 6 seconds or less (Performance)
A2. Run for a sustained amount of time without major errors on an Emulator running Gremlins (Reliability)
A3. Have robust error handling (Reliability, Usability)
A4. Allow users to cancel I/O operations (Usability)
A5. Comply with general Palm user interface standards (Conformance to standards, Usability)
A6. Have user friendly error reporting (Usability)

2.1.3 Constraints

C1. Support WQV-3 and WQV-10 watches (Portability)
C2. Must operate on Palm OS 3.1-4.1 (Portability)
C3. Display images using the highest bit depth and display resolution available on the device, independent of the support from the OS (Portability)
C4. Support high resolution Sony CLIÉ devices (Portability)
C5. Must work on devices with 1-bit or 2-bit as the highest depth available (Portability)
C6. Use Jog Dial on CLIÉ devices (Portability, Usability)
C7. Should be able to export an image to a PAMD Plug-In (Integrability)

2.2 PC Application

This section lists the functional requirements, architectural requirements, and constraints for the PC application.

2.2.1 Functional Requirements

R1.   View the images stored in the Palm application's database after a HotSync (Interoperability)
R2.   Import JPEG images from the Palm application (Interoperability)
R3.   Export JPEG images to the Palm application (Interoperability)
R4.   Automatically scale images to meet the watch camera specification for JPEG images

2.2.2 Architectural Requirements

A1.   Interface with the Palm HotSync manager (Integrability)
A2.   Have robust error handling (Reliability, Usability)
A3.   Have user friendly error reporting (Usability)

2.2.3 Constraints

C1.   Must be compatible with Win 9X, Win 2K, Win ME, and Win XP (Portability)
C1.   Must support 8, 16, 24, and 32 bit color depths (Portability)
C1.   Must run on all hardware running Win 9X, Win 2K, Win ME, and Win XP (Portability)

3   Utility Tree

The scenarios are organized and prioritized in a utility tree. The scenarios are represented as "leaves" in the utility tree. In addition to a scenario, each leaf contains indications of the importance of the scenario and the risk associated with the scenario. This is represented in the diagram as (*importance*, *risk*) where the values for *importance* and *risk* are High (H), Medium (M), or Low (L). Descriptions of the scenarios can be found in Appendix B.

3.1   Palm Application

The scenarios below outlined in red are the scenarios that have the highest importance and risk. It can be clearly seen that portability is the most important quality attribute for the Palm application. Performance and interoperability are also important, while reliability and usability are less important.
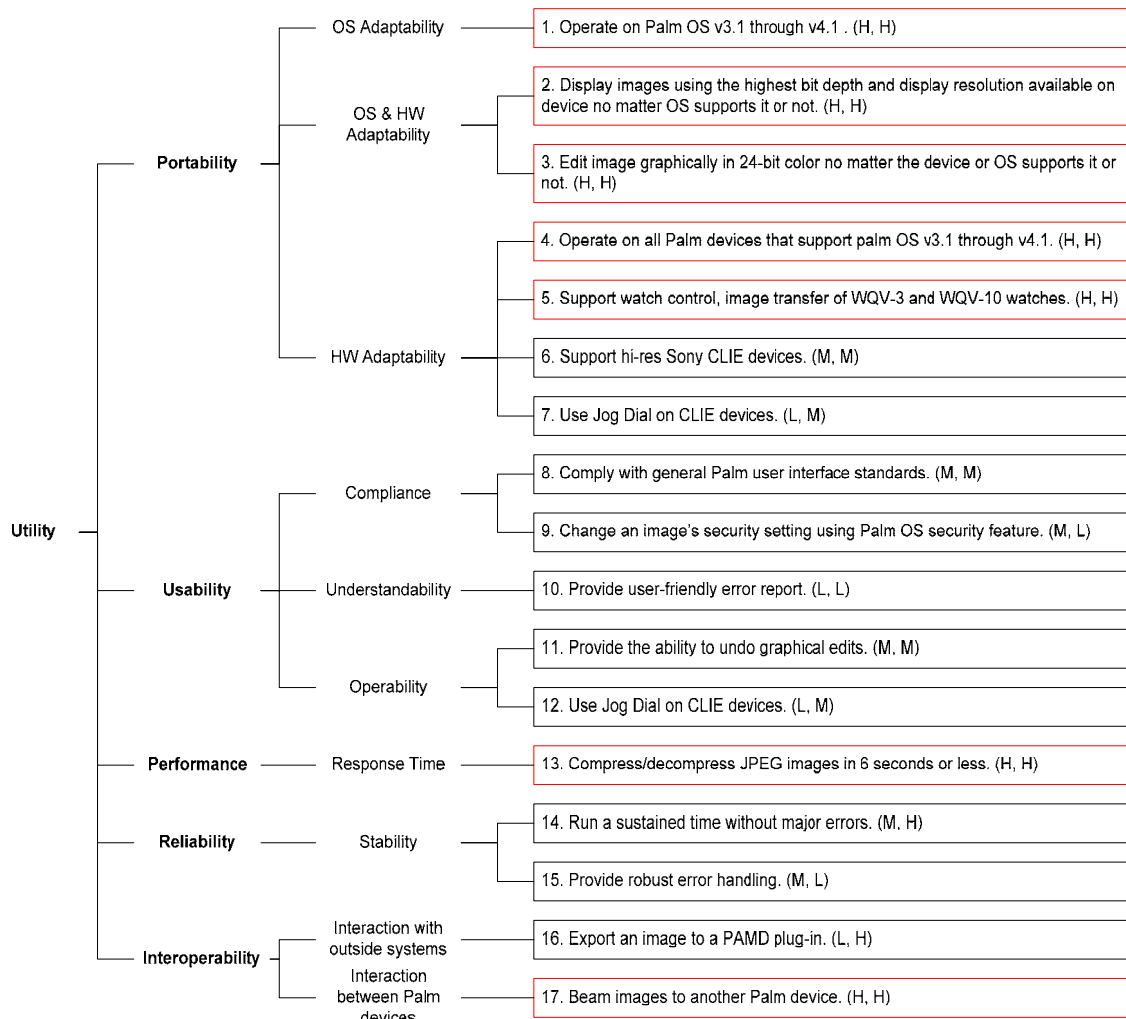
Figure 2: Utility tree for the Palm application

3.2 PC Application

From the outlined scenarios it can be seen that interoperability is the most important quality attribute for the PC application.
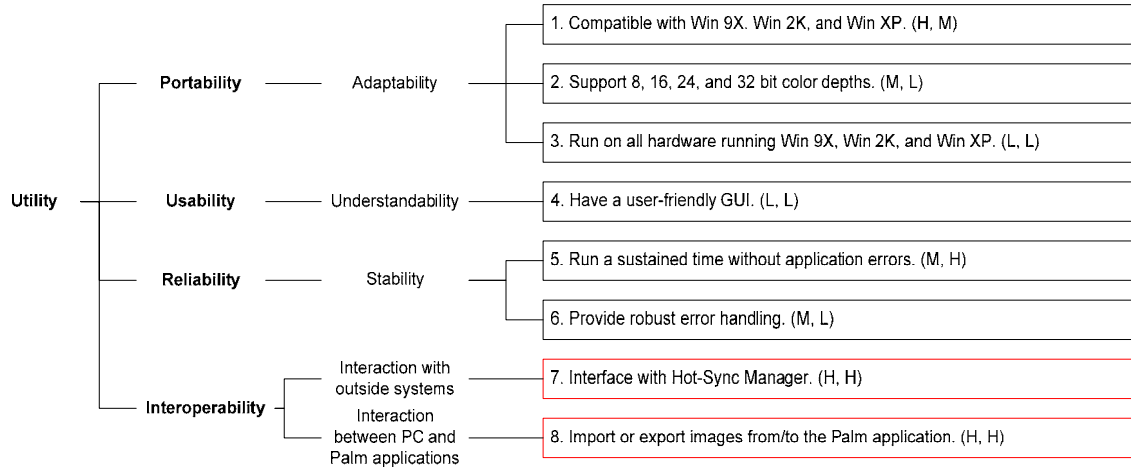
Figure 3: Utility tree for the PC application

4 System Architecture

The following sections discuss the proposed architecture for the WQV companion applications. The first architecture diagram shows both the PC application and the Palm application. The rest of the diagrams and the rest of the document will focus on the Palm application.

4.1 Component and Connector View – Overall System

The diagram below shows a high-level view of the interactions within WQV-Buddy and the interaction between WQV-Buddy and the external watch system. This view of the system addresses the interoperability requirements of the PC and Palm applications. It is meant to be a context diagram of the system and to show the communications required between various components. The specific components and connectors are discussed below. The ports shown in the diagram represent the communication interfaces implemented by the components. The details of the interface are not discussed and represented here.
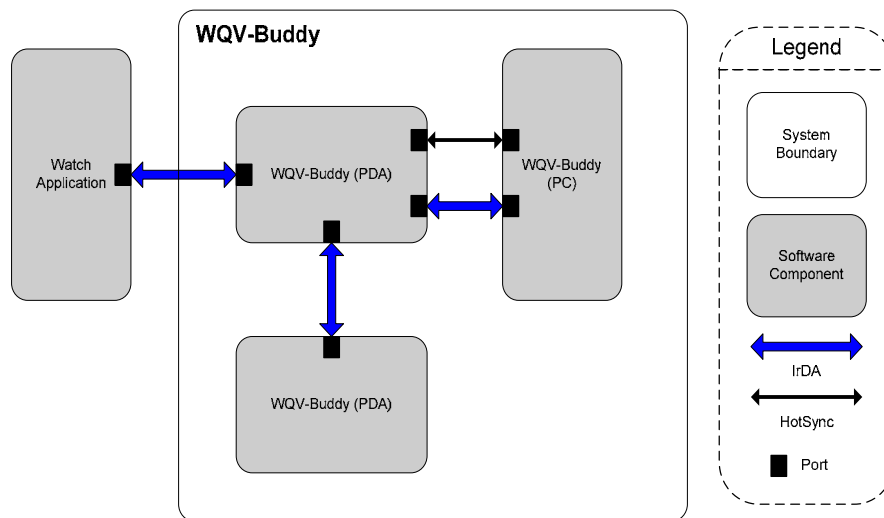
Figure 4: WQV-Buddy component and connector view

Watch Application
The watch application is an external component to the WQV-Buddy system. In order to satisfy the requirements of controlling the watch and transferring images to/from the watch, the Palm (PDA) application must work with the software application on the Casio watch. The means of communication between the two components is IrDA infrared communication.

WQV-Buddy (PDA)
The WQV-Buddy (PDA) components show the Palm application at run-time. The executing applications must communicate with the external watch application (as discussed above), the executing PC application, as well as with each other. Communication with the PC occurs in two ways, through HotSync or IrDA communication. The two components communicate with each other using IrDA infrared communication.

WQV-Buddy (PC)
The WQV-Buddy (PC) component is the PC application at run-time. It must communicate with the executing Palm application as discussed above.

IrDA
The IrDA connector describes the interaction between any two components with infrared support. It is a communication protocol of transferring data over the infrared link.

HotSync
The HotSync connector describes the interaction between the Palm component and the PC component. It is a method of exchanging data between a handheld and a Computer.

4.2   Component and Connector View – Palm Application

The component and connector view of the proposed architecture is shown below. The architecture is a tiered architecture that contains two tiers. The details of these tiers are discussed below.
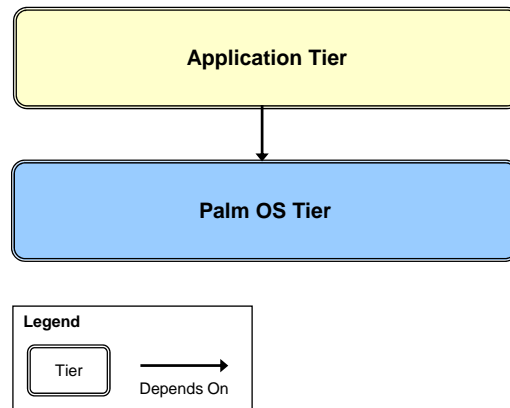
Figure 5: Palm application component and connector view

Application Tier
The application tier is the executing Palm application. This tier is responsible for providing a user interface, for executing the business logic of the application and for handling device dependent and operating system dependent functionalities. It should be noticed that the application tier depends only on the Palm OS tier. There are no other dependencies for the application.

Palm OS Tier
The Palm OS tier is the executing Palm operating system. The ability of the Palm application to execute relies on the existence of this operating system.

4.3   Module View – Palm Application

The following diagram shows the module view of the WQV-Buddy architecture. Because portability is the system's driving quality attribute, the layered architecture style has been selected. The current system consists of five layers, which are the User Interface Layer, the Application Layer, the Portability Layer, the Palm OS, and the Palm Hardware. Although the Palm OS and the Palm Hardware are outside the boundary of the designed system, they are kept in the diagram to show the dependency relationship among the designed layers and their environment.
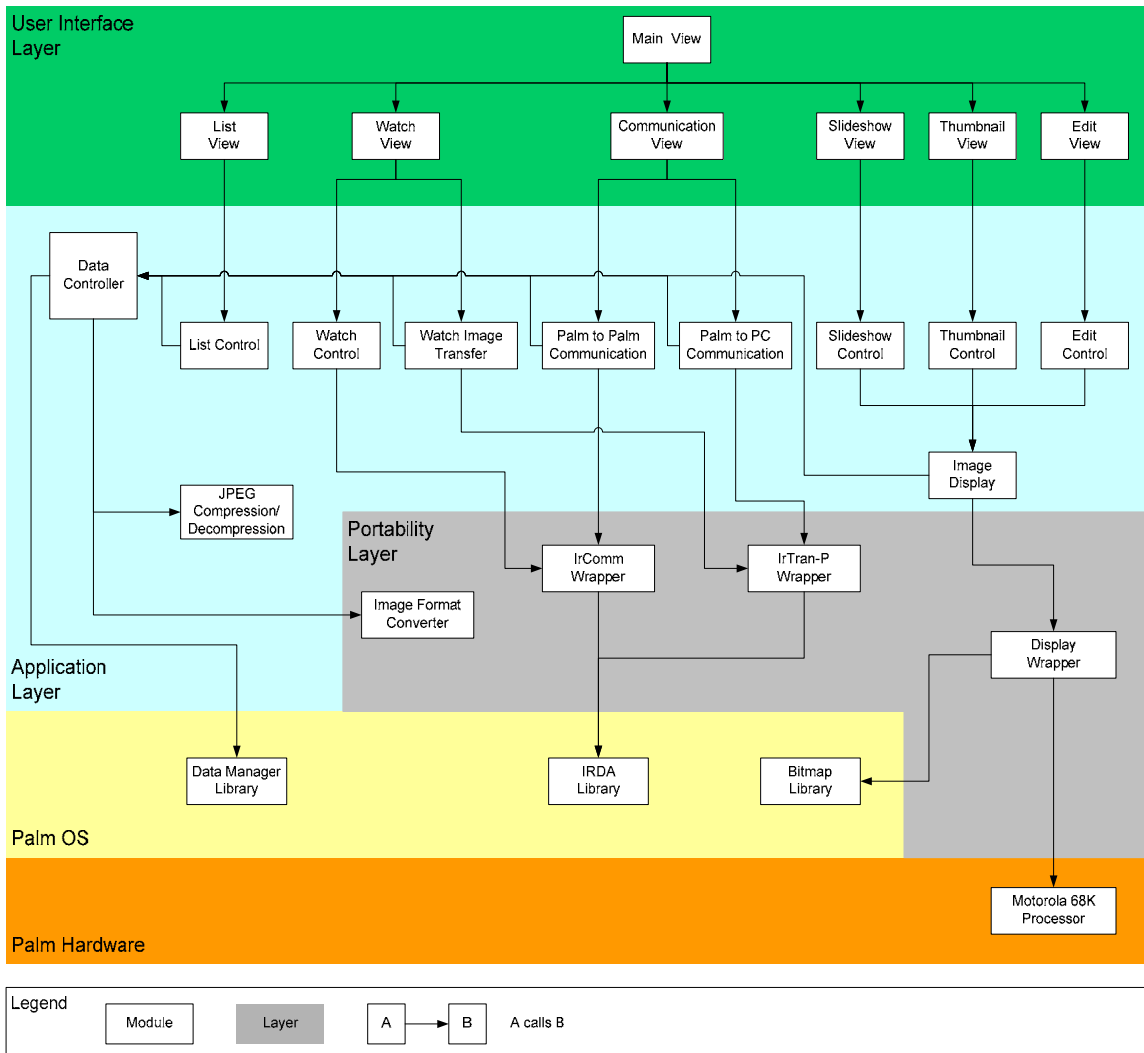
Figure 6: Palm application module view

User Interface Layer

The user interface layer provides the user access to WQV-Buddy's functions. This layer is introduced to separate the business logic control from the user interface. This isolates the usability issues and helps to divide the implementation efforts.

- *Main View* – the central control of the user interface layer. All of the other UI components (views) are directly connected to the Main View and launched by it upon request from the user.
- *List View* – displays the list of existing images in the database. The user can select any one of the images in the list by clicking on the image's name.
- *Communication View* – provides access to infrared communications with other computing devices, such as PCs and Palms. One of the operations that can be executed from this view is transferring an image from the watch.
- *Slideshow View* – displays the images as a slideshow.
- *Thumbnail View* – displays the thumbnail view of images.
- *Edit View* – provides access to the image editing functions.
- *Watch View* – provides access to the functions that control the Casio watch and upload/download images from the Casio watch.

Application Layer

This layer contains all of the business logic of the application. It will receive the user's command from the user interface layer and perform operations based on the user's request.

- *List Control* – contains the routines to list the images in the database.
- *Watch Control* – contains the routines to control the watch remotely
- *Watch Image Transfer* – contains the routines to transfer images to and from the Casio watch.
- *Palm to PC Communication* – contains the routines to transfer images to and from a PC.
- *Palm to Palm Communication* – contains the routines that are used to beam images from one Palm to another.
- *Slideshow Control* – contains routines to display images as a slideshow.
- *Thumb Control* – contains routines to display images as thumbnails.
- *Edit Control* – contains image editing routines, such as drawing a line or a circle. It is also required to use 24-bit color regardless of the available color-depth on the Palm.
- *Data Controller* – contains the routines that wraps the functionalities for manipulating the Palm database, such as retrieving the list of all images.
- *JPEG Compression/Decompression* – used for compressing bitmap images to JPEG format and for decompressing JPEG images to bitmap format. This is used when images are imported from the watch or are exported to the watch. It is also used when transferring images to a PC or to another PDA,
- *Image Display* – contains routines to display an image on the Palm device.

Portability Layer

This layer is used to wrap the functionalities that vary across different Palm device hardware and OS versions. It will hide the Palm platform variations from the Application Layer.

- *Image Format Convert* – provides the bitmap depth conversion to the format that is optimized for the Palm platform.

- *IrComm Wrapper* – provides a platform independent interface to the PDA infrared interface.
- *IrTran-P Wrapper* – provides an interface to the IrTran-P interface regardless of Palm OS support.
- *Display Wrapper* – provides display capabilities that utilize the specific device hardware capabilities.

Palm OS

This layer is the operating system part of the Palm device. Although it offers a lot of useful functionalities, the functionalities vary based on OS version. All the code modules (or libraries) shown here are supported by all versions of the Palm OS.

- *Data Manager Library* – required to manage all data storage related activities on the Palm OS. This will be mainly used by the Data Controller.
- *IrDA Library* – required for all modules that use infrared communications on the Palm device, such as the IrComm and IrTran-P wrappers.
- *Bitmap Library* – required for image related operations, such as JPEG Compression/Decompression, format conversion, and image display.

Palm Hardware

This layer is essentially the hardware of the Palm device. It is outside of our system and are not to be modified by the application. However, some application modules need to bypass the Palm OS and interact with the hardware's firmware directly to fulfill some requirements.

- *Motorola 68K Processor* – the hardware that needs to be accessed to utilize the display capabilities of the hardware independent of support from the OS.

5   Architectural Alternative

The following sections discuss the main alternative to the proposed architecture. From the results of the requirements and scenario prioritization, it is determined that portability is the most important quality attribute for the PC application. There are two main approaches to portability considered. The first approach, using a wrapper layer, is shown in the proposed architecture above. The second approach is using a virtual machine tier and is discussed below.

5.1   Component and Connector View

The component and connector view of the alternative architecture is shown below. The architecture is a tiered architecture that contains three tiers which are discussed below. This architecture differs from the proposed architecture by introducing a tier between the executing Palm application and the Palm OS.
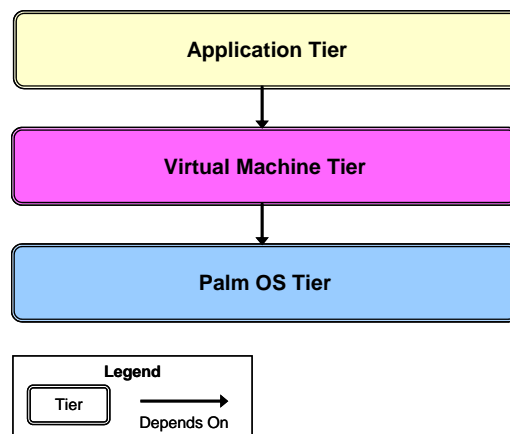
Figure 7: Palm application component and connector view

<u>**Application Tier**</u>
The application tier is the executing Palm application. This tier is responsible for providing a user interface and for executing the business logic of the application. This application tier differs from the application tier mentioned above because it does not handle device dependent and operating system dependent functionalities. These functionalities are now handled by the virtual machine tier. The execution of the application tier is dependent upon the execution of the virtual machine tier.

<u>**Virtual Machine Tier**</u>
The virtual machine tier is an executing application that separates the application from the operating system. The purpose of introducing this tier is to increase the portability of the Palm application. The application does not have to be recompiled or modified in order to be executed on devices with various Palm OS and device capabilities.

<u>**Palm OS Tier**</u>
The Palm OS tier is the executing Palm operating system. The ability of the virtual machine to execute relies on the existence of this operating system.

5.2 Module View

The following diagram shows the module view of the alternative architecture. Like the module view of the proposed architecture, this view shows a layered architecture. This architecture, however, only contains two layers. These are the User Interface Layer and the Application Layer. The details of these layers are discussed below.
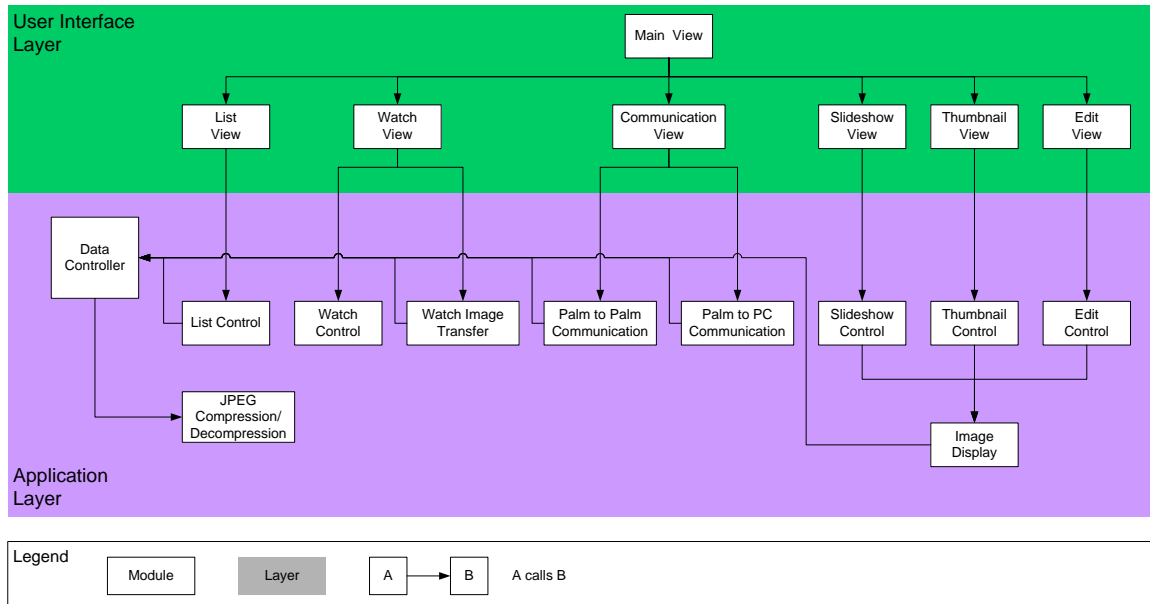
Figure 8: Palm application module view

<u>**User Interface Layer**</u>
The user interface layer will provide the user access to WQV-Buddy's functions. This layer is introduced to separate the business logic control from the user interface. This isolates the usability issues and divides the implementation efforts. The modules in this layer are the same as those in the proposed architecture. See the discussion of the proposed architecture for details about these modules.

<u>**Application Layer**</u>
This layer contains all of the business logic of the application. It will receive the user's command from the user interface layer and perform operations based on the user's request. The difference between this layer and the application layer in the proposed architecture is that this layer does not make calls to a wrapper layer in an effort to support portability. The layer's modules contain only business logic calls that will be interpreted by the virtual machine at run-time. Again, a discussion of the individual modules can be found in the proposed architecture description above.

The module view of the virtual machine is not shown here. It is assumed that the virtual machine will either be a third party product or will be developed separately.

6   Scenario Analysis

The following sections contain detailed analysis for selected scenarios from the Palm application utility tree. Two scenarios are chosen that are representative of the two most important quality attributes of the system, portability and performance.

6.1   Scenario 1: Operate on Palm OS v3.1 through v4.1.

Attribute: Portability/ OS adaptability
Stimulus: Users run the same application on different PDAs running any version of the Palm OS (version 3.1 through 4.1).
Response: All of the functionalities run correctly to the extent possible on all of the PDAs.

Table 1: Analysis for scenario 1.

Architectural Decisions	Risks	Sensitivity Points	Tradeoffs
Wrapper layer architecture	1	1	1
Virtual machine architecture	2	2, 3	2

Risks:
1. The development time of the application may be increased as a result of having to implement the wrapper layer. This may have an impact on the schedule of the project.
2. The dependence on an independently developed virtual machine may have an impact on the ability to implement all of the desired functionalities. There may be little or no control over the implementation of the virtual machine.

Sensitivity Points:
1. Portability is increased because of the existence of the wrapper layer. Adding support for new platforms or operating systems is separated into this layer and does not affect the other layers.
2. Portability is increased because of the virtual machine tier. Support for various operating systems and hardware is hidden from the application.
3. Modifiability is increased because changes to the functionalities in the user interface or application layers will not require adding extra operating system or hardware support.

Tradeoffs:
1. There is a tradeoff between portability and modifiability. Portability is increased by separating portability concerns into a wrapper layer. Modifiability is decreased because of the potential of having to add support in the wrapper layer when new functionality is added in the upper layers.
2. There is a tradeoff between modifiability and functionality. Modifiability is increased because functionality can be added without being concerned with portability issues. Functionality can be decreased as a result of not having control over the implementation of the virtual machine.

Reasoning:
The wrapper layer architecture is selected as a result of the tradeoffs between the two architectures. Both architectures support portability, but the virtual machine architecture does so at a potential loss of functionality. Since loss of functionality is less desirable than reduced modifiability, the wrapper layer architecture is chosen.

6.2 Scenario 13: Compress/decompress JPEG images in 6 seconds or less.

Attribute: Performance/ Response Time
Stimulus: User imports a JPEG image from the Casio watch and displays it. User uploads an image in Bitmap format to the Casio watch that only accepts JPEG format.
Response: The image is compressed/decompressed within 6 seconds.

Table 2: Analysis for scenario 13.

Architectural Decisions	Risks	Sensitivity Points	Tradeoffs
Wrapper layer architecture	1	1	1
Virtual machine architecture	2	2	2

Risks:
1. The development time of the application may be increased as a result of having to implement the wrapper layer. This may have an impact on the schedule of the project.
2. The virtual machine may result in decreased execution speed of the Palm application. It may not be possible to ensure that the virtual machine is implemented efficiently.

Sensitivity Points:
1. Performance is increased because of the exceptions made to the layered style. Modules in the application layer use functions provided by the Palm OS layer directly.
2. Performance is decreased because all functions in the application tier must go through the virtual machine tier to be executed.

Tradeoffs:
1. There is a tradeoff between performance and modifiability. Performance is increased because of the exception made to allow the application layer to bypass the wrapper layer. This, however, reduces modifiability, since the coupling between the application layer and Palm OS layer becomes tighter.
2. There is a tradeoff between portability and performance. Portability is increased because support for various operating systems and hardware is hidden in the virtual machine tier. Performance is decreased as a result of having all functions interpreted by the virtual machine tier.

Reasoning:
The wrapper layer architecture is selected because performance is a higher priority than modifiability. It is shown in the analysis for scenario 1 that both architectures support portability. Since the wrapper layer increased performance while the virtual machine decreased performance, the wrapper layer architecture is chosen.

7 Potential Extensions

There are several possible extensions that could be made to the system. The extensions could be made by future MSE teams that want to extend the functionality of the application. Other developers that want to use the software with future versions of the Palm OS or other Palm-based hardware could also extend the system as needed. Some of the possible extensions are listed below along with their architectural implications.

- **Port the application on future versions of Palm OS (beyond Palm OS 4.1):** This would be easy with the current architecture. The main concern here is the portability of the application. The OS dependent functionalities are wrapped into the portability wrapper layer. Future versions of Palm OS will follow the policy of providing backwards compatibility. Therefore, all changes can be restricted to the wrapper layer.

- **Support new Palm hardware:** This affects the portability of the application. Each Palm-based PDA vendor tries to be different from others by including new "cool" features, e.g. Sony CLIÉ uses a Jog-Dial for scrolling and selection. Support for these "cool" features can be isolated into the wrapper layer.

- **Support new versions of PC operating systems:** The PC companion application of WQV-Buddy, makes all calls using the Conduit library provided by the Palm OS. Other calls rely on the Win32 API. Hence the new versions of PC operating systems (Windows) will have no problems running the PC application if the new API is backward compatible with the Win32 API.

- **Use features provided by other plug-ins (like PAMD):** This is a test of the integrability of WQV-Buddy. There exist other plug-ins that are able to provide more utility for the images by providing additional image editing, processing features, and other functionality. Such plug-ins usually interact with the database. WQV-Buddy isolates the database from rest of the application using the database interface module. Any new plug-in can be easily integrated with WQV-Buddy by interfacing with the database interface.

8 ATAM Process Evaluation

The team followed a continuous architecture evaluation process along with architecture development. The quality attributes and the driving business goals were identified before going ahead with the architecture. A vertical use case was prepared that covers most of the application's functionalities. This use case was used to derive the higher level architecture for the system. The attribute prioritization was done using the utility tree. The whole process was very helpful in confirming that the chosen architecture was indeed the most suitable one.

Before the ATAM process was initiated, the team had already identified the stakeholders for getting a clearer idea of the requirements at the beginning of the project. The clients and the users were identified as the most important stakeholders. The business and timing constraints were stated by the client. The usability constraints were stated by the Palm programming guidelines document. The other constraints were clearly stated in the requirements document.

8.1 Identification of driving quality attributes

The team began by identifying the quality attributes of the project. The business drivers were identified after having a meeting with the client. These were then used to identify the driving quality attributes.

The team then prioritized the quality attributes. The level of difficulty in implementing the required features of the system was used to set the priority among the quality attributes. Most of the requirements required portability across the different versions of Palm OS. Hence portability was identified as the most important. The utility tree was then derived from these attributes.

8.2 Testing the architecture against the quality attributes

To test the derived architecture against these quality attributes, the team came up with an alternative architecture. This architecture used a virtual machine tiered style. Although this was not the best architecture, it helped to verify that the layered architecture supports the important quality attributes. It also helped to identify risks and tradeoffs in the architecture design.

8.3 Conclusion

Portability was clearly the most important driving attribute of the application, and it was very obvious from the beginning of the project. A layered architecture with layers bypassing the nearby layers for the sake of performance (overlays) was adopted early on by the team. Hence the ATAM process was not useful to the team in the traditional sense. The process was useful in creating the detailed architecture of the system. It helped to identify some of the flaws in the identified components and in the ways they were connected. It also helped to clarify the functionality of each layer in greater detail and to identify each connector.

Appendix A: Project Requirements

Palm OS application

The purpose of the Palm application is to allow for the transferring of images singly or in a group from the Casio WQV-3 and WQV-10 watches to a PDA and back. In addition, the application will allow editing of images, and it will allow a user to remotely control the Casio watch. It will also allow the grouping of images by categories.

Functional Requirements

The Palm application must:

1. Control the watch remotely
    a. Set date and time
    b. Advance one image on the watch (forward and backward)
    c. Fast scroll through images on the watch (forward and backward)
    d. Send a quit message to watch
    e. All I/O operations should be cancelable by the user from either the watch or PDA

2. Transfer images to and from the watch
    a. Receive all images from watch
    b. Send category of images to watch
    c. Send or receive a selected image
    d. All I/O operations should be cancelable by the user from either the watch or PDA

3. View images
    a. List view -- display the image's time/date and picture description in a list (description can be truncated)
        i. Support manual sorting of images in this mode
        ii. Ability to select a description to display the full image, or send the image to a watch or PC
    b. Thumbnail view -- display thumbnails of images including title/date time and picture description (description can be truncated)
        i. Ability to scroll through thumbnails quickly (no wait except for display)
        ii. Ability to select a thumbnail to display the full image or to send the image to a watch or PC
    c. Slide show view – sequence through all the images in a category (category can be all) displaying each image at the user specified sequence rate (rate can be menu selectable)
        i. Ability to start and stop the slide show
        ii. Provide eight different drawing effects to display the image (the eight effects should be of the team's choosing – ex: line fill, slide)
        iii. Ability to specify that the drawing effects should be random during the slide-show
    d. Sort images by date/time/title, title/date/time or manually.
        i. Sorting should happen automatically for the date/time/title or title/date/time option

4. Edit images
   a. Edit view – displays the full size image (image scrolling needed)
      i. Edit the image graphically (must be capable of drawing on an image in a specified color)
      ii. Graphical editing must be done in 24-bit color even if the display does not support this color depth
      iii. Provide undo facility for a graphical change to the image until the image is saved
      iv. Change the image date, time, or picture description
      v. Create a new image using a specified background
      vi. Change the image's security settings
      vii. Change the image's category
   b. Allow deletion of a single image or of all the images in a category

5. Communicate with a PC over IrTran-P
   a. Send a selected image to the PC
   b. Send a category (category can be all) of images to the PC
   c. All I/O operations should be cancelable by the user

6. Beam images to other Palm devices loaded with the same application
   a. Beam a selected image
   b. Beam a category (category can be all) of images

7. Support category operations
   a. Edit, create and delete categories
      i. When a category is deleted, images in that category go to "all" category
   b. View only images in a specific category (category can be all) (applicable to any view mode)
   c. Ability to assign images to categories

8. Secure data using the Palm OS security feature
   a. Mark images as private preventing them from being viewed without a password
   b. Show, mask, or hide private records
   c. Prompt for password to display private records
   d. Automatically delete private records when the password is lost

9. Export an image to a PAMD Plug-In

Non-Functional Requirements

Performance

1. Image compression / decompression should take 6 seconds or less (this includes decompression during image display and any other image conversions)

Reliability

2. Should run a sustained time (30 seconds) without major errors on the Palm emulator running Gremlins

Interface

Should have robust error handing with user friendly reporting

1. Should comply with the general Palm user interface standards

Constraints

Hardware Constraints

1. Must be compatible with the WQV-3 and WQV-10 watches – must automatically detect and communicate with WQV-3 and WQV-10 (slight protocol differences)

2. Must display images using the highest color/grayscale bit depth and display resolution available on the device (includes support for the hi-res Sony CLIÉ devices)

3. Must make appropriate use of the Jog Dial on CLIÉ devices

4. Must display images using at least 4-bit grayscale on Palm devices using the Dragonball EZ processor and 4-bit display regardless of OS support.

5. Must work on devices that have hardware that only supports 2-bit or 1-bit depths in the highest bit depth available

6. Must work with Win 2K and Win XP which support receiving IrTran-P images

Software Constraints

1. Must operate on Palm OS version 3.1 through 4.1

PC Application

The PC application is a companion to Palm application.

Functional Requirements

The PC application must:

1. View the images stored in the Palm applications' database(s) after a HotSync
   a. Display images
   b. Display date/time and picture description for the images

2. Select images stored on the Palm and import them to the PC as JPEG images
   a. Select the import directory

3. Export selected JPEG images to the Palm application
   a. Automatically scale images to meet the Wrist Camera specification for JPEG images
   b. Automatically place images to be exported into the users install folder
   c. Import images manually under user direction

Non-Functional Requirements

Interface

1. Should have robust error handing with user friendly reporting

2. Should interface with the Palm HotSync manager anywhere that HotSync works

Constraints

Software Constraints

1. Must be compatible with the Win 9X, Win 2K, Win ME, and Win XP platforms

2. Must support 8, 16, 24, and 32 bit color depths

Hardware Constraints

1. Must run on all hardware running Win 9X, Win 2K, Win ME, and Win XP

Appendix B: Scenarios

In order to determine the quality goals against which the architecture will be judged, scenarios are used. A scenario is a short statement that describes an interaction of a stakeholder with the system. The scenarios for the Palm and PC applications are given below.

Palm Application

Scenario 1: Operate on Palm OS v3.1 through v4.1.

Attribute: Portability/ OS adaptability
Stimulus: Users run the same application on different PDAs running any of the Palm OS (version 3.1 through 4.1).
Response: All the functionalities run correctly to the extent possible on all the PDAs.

Scenario 2: Display images using the highest bit depth and display resolution available on device no matter OS supports it or not.

Attribute: Portability/ OS and hardware adaptability
Stimulus: Users run application on a Palm V running Palm OS 3.1. The device has a display capable of 4-bit grayscale. Palm OS 3.1 does not support this feature.
Response: The application displays images at 4-bit grayscale which is the color depth that is possible on the device.

Scenario 3: Edit image graphically in 24-bit color no matter the device or OS supports it or not.

Attribute: Portability/ OS and hardware adaptability
Stimulus: Users edit an image on a Palm V running Palm OS 3.1. The device supports grayscale displays.
Response: The image is edited in 24-bit color, and saves in that format.

Scenario 4: Operate on all Palm devices that support palm OS v3.1 through v4.1.

Attribute: Portability/ Hardware Adaptability
Stimulus: Users run the same application on different PDAs running any of the Palm OSes version 3.1 through 4.1, from any of the manufacturers like Sony, Handspring, Palm Inc, etc.
Response: All the functionalities run correctly to the extent possible on all the PDAs.

Scenario 5: Support watch-control and image transfer from WQV-3 and WQV-10 watches.

Attribute: Portability/ Hardware Adaptability
Stimulus: Users use the WQV-3 instead of WQV-10 with the application.
Response: The users are able to control the watch and transfer images from it the same way as it is able to do with the WQV-10.

Scenario 6: Support hi-res Sony CLIE devices.

Attribute: Portability/ Hardware Adaptability
Stimulus: Users run the application on high resolution Sony CLIÉ devices
Response: The application utilizes the high resolution on the devices while displaying images and other functionalities.

Scenario 7: Use Jog Dial on CLIÉ devices.

Attribute: Portability/ Hardware Adaptability
Stimulus: Users want to use the Jog Dial navigator when running the application on Sony CLIÉ palm devices.
Response: The application's user interface responds to the Jog Dial navigator events appropriately. Such events include,
- Press the Jog Dial navigator down and then release.
- Rotate the Jog Dial clockwise.
- Rotate the Jog Dial counter-clockwise.

Scenario 8: Comply with general Palm user interface standards.

Attribute: Usability/ Compliance
Stimulus: A regular Palm user that is new to our application can use the system without any pre-knowledge.
Response: The application works the way Palm users expect it to work, that is, it looks and behaves like other Palm OS applications. A new user with Palm experience should be able to use the application without referring to any manual.

Scenario 9: Change an image's security setting using Palm OS security features.

Attribute: Usability/ Compliance
Stimulus: The user wants to mark records as private, so that they may be shown only when the user enters the correct password in the device's Security application.
Response: The application shows/masks/hides the private records appropriately based on the current security preference on the Palm device. For example, if the user has assigned a password in the system's Security application and has chosen to hide private records, any image marked private in WQV Buddy does not show up in any view.

Scenario 10: Provide a user-friendly error report.

Attribute: Usability/ Understandability
Stimulus: An error occurs.
Response: The application provides appropriate and friendly error report that let the user understand which happened (including which action caused error or could not be completed) and do the correct operation.

Scenario 11: Provide the ability to undo graphical edits.

Attribute: Usability/ Operability
Stimulus: The user makes unwanted graphical edits.
Response: The system should allow the user to reverse any unwanted edits.

Scenario 12: Use Jog Dial on CLIÉ devices.

Attribute: Usability/ Operability
Stimulus: Users expect the Jog Dial navigator to influence the application in similar ways as in other Sony CLIÉ palm applications.
Response: The application's user interface responds to the Jog Dial navigator events appropriately as other Sony CLIÉ palm applications.
- Event "Press the Jog Dial navigator down and then release" should be mapped to "Enter" function.

- Event "Rotate the Jog Dial clockwise" should be mapped to "Increase" or "Scroll down" functions.
- Event "Rotate the Jog Dial counter-clockwise" should be mapped to "Decrease" or "Scroll up" functions.

Scenario 13: Compress/decompress JPEG images in 6 seconds or less.

Attribute: Performance/ Response Time
Stimulus: User imports a JPEG image from the Casio watch and displays it. User uploads an image in Bitmap format to the Casio watch that only accepts JPEG format.
Response: The image is compressed/decompressed within 6 seconds.

Scenario 14: Run a sustained time without application errors

Attribute: Reliability/ Stability
Stimulus: Users can continually running the application for 2 hours
Response: There is no fatal error occurred during the users' operations. The fatal error defined here requires the soft or hard reset to restore the system.

Scenario 15: Provide robust error handling

Attribute: Reliability/ Stability
Stimulus: The normal errors occur during the user operations.
Response: The application can catch the errors as exceptions and report to the user friendly. The application should continue to work properly without any negative effects.

Scenario 16: Export an image to a PAMD plug-in

Attribute: Interoperability/ Interaction with outside systems
Stimulus: The application exports an image to PAMD plug-in directly
Response: The application exported the image to PAMD plug-in correctly.

Scenario 17: Beam images to another Palm device.

Attribute: Interoperability / Interaction between Palm devices
Stimulus: The user wants to share the images with another Palm user by beaming the images.
Response: The application transfers selected image(s) to another Palm that running the same application via infrared port.

PC Application

Scenario 1: Compatible with Win 9X, Win 2K and Win XP

Attribute: Portability/ Adaptability
Stimulus: The PC application is installed on different windows operating systems, which includes Win 98, Win 2000 and Win XP.
Response: The application can work properly on different operation systems with the same set of functions for the users.

Scenario 2: Support 8, 16, 24 and 32 bits color depths display

Attribute: Portability/ Adaptability
Stimulus: The PC application is run with different display depths: 8, 16, 24 and 32.

Response: The PC application can display the images properly on different display depths without taking any additional configuration operations.

Scenario 3: Should run on all hardware running Win 9X, Win 2K, and Win XP.

Attribute: Portability/ Adaptability
Stimulus: The application is installed and run on devices that are able to run Win 9X, Win 2K, and Win XP.
Response: All of the application functionalities should work correctly on each device.

Scenario 4: Should have a user-friendly GUI.

Attribute: Usability/ Understandability
Stimulus: A regular Window's user that is new to our application uses the application.
Response: The application works the way Window's users expect it to work, that is, it looks and behaves like other Windows applications. A new user that has Window's experience should be able to use the application without referring to any manual. The application should also be easy for expert users of our application to use.

Scenario 5: Run a sustained time without application errors.

Attribute: Reliability/ Stability
Stimulus: The application is used for 2 hours consecutively.
Response: No errors occur.

Scenario 6: Provide robust error handling.

Attribute: Reliability/ Stability
Stimulus: An error occurs.
Response: The error is handled appropriately such that the application does not hang or crash.

Scenario 7: Interface with HotSync manager.

Attribute: Interoperability/ Interaction with outside systems
Stimulus: The user wants to synchronize data between the Palm application and the PC application.
Response: The data is synchronized correctly.

Scenario 8: PC application imports or exports images from/to the Palm application.

Attribute: Interoperability/ Interaction between PC and Palm applications
Stimulus: The user wants to share image between the Palm application and the PC application.
Response: Image is transferred from/to the Palm application.