Master of Science in Internetworking

32932 PROJECT A

Project Proposal

Student Name: Paul McIntosh

Student Number: 9901 9248

Due Date: 01/06/2001

Supervisors: Paul Kennedy, David Wilson

1.0 Title and Thesis

Title: "Developing Infrared Distributed Applications on the Windows CE 3.0 Platform Using Rational Rose RealTime"

Thesis: Extending Rational Rose RealTime to enable development of distributed models, which utilise infrared communications between Windows 2000 (PC) and Windows CE 3.0 (Pocket PC) targets.

2.0 Thesis Description

Background

Rational Rose RealTime (Rose RealTime) is a visual modelling software development application produced by Rational Software. Rose RealTime utilises Unified Modelling Language (UML) and real-time extensions to design real-time embedded applications, generate code from the design, compile, debug and deploy, all from within the one tool.

Rose RealTime also has the ability, through the use of another Rational product called Connexis, to apply the same visual modelling development principles to distributed applications.

For a distributed application, and for Rose RealTime to communicate with a target, there needs to be a communications medium. Windows CE 3.0 on the Compaq iPAQ Pocket PC (Pocket PC) and Windows 2000 on a standard PC have compatible infrared communications functionality provided through Infrared Data Association (IrDA) standard ports.

Rose RealTime requires configuration to target specific platforms. Supported out-of-the-box platforms currently include Windows NT as host and target, and Windows CE 3.0 for the target SH3 processor. At present Rose RealTime and Connexis have not been configured (ported) to the target Windows CE 3.0 Pocket PC platform, which uses an Intel StrongArm processor. In addition Rose RealTime 2001a.04.00, the version that supports the SH3 target, was only released days ago (28/05/2001) and therefore there has been no investigation into Windows CE 3.0 and infrared in general.

Description

The primary goal of this thesis is to develop a proven road map for the use of Rose RealTime for the development of distributed models across Windows PC and Pocket PC platforms using infrared. The secondary goal is the discovery of specific and generic "real world" problems which surface during this process, documenting the problem and solution (if applicable).

This project proposal indicates a proposed development road map and the ideal expected results in the form of software and processes. The project will execute this road map and attempt to meet the ideal expected results.

It is anticipated though, that during execution, the road map will evolve as a result of constant evaluation and revision. At the completion, a revised road map will be produced which either utilises the software, processes and knowledge developed in the first execution or outline the blocking issues ("dead ends" discovered) that need to be overcome.

Ideal expected results:

Software

1. Rose RealTime port to Windows CE 3.0 for Pocket PC (recompiled runtime service library, build scripts and example application)
2. Example application, which demonstrates an infrared distributed application
3. Reusable components for developing infrared distributed applications
4. Any additional software which may be required to extend Rose RealTime/Connexis to more effectively use infrared as a development communication medium.

Processes

1. Road Map for creating a target deployment package for Windows CE 3.0 on Pocket PC using Microsoft's eMbedded Visual C++ 3.0
2. Road Map for developing infrared distributed applications using Rose RealTime and reusable components.
3. Configuration guide to using infrared as a development communication medium.

Possible blocking issues:

With any software development project, especially when integrating development tools in a way that has not been done before, the ideal expectations rarely occur as planned. The discovery of any blocking issues is just as important as the ideal expected results. In the event of issues that block the development process, the secondary goal of this thesis is to discover if there are workarounds and possible solutions. Some of the possible blocking issues are:

- Conflicts in naming conventions between Rose RealTime code and eMbedded Visual C++.

- Infrared, due to being half-duplex or simply to difficult to implement, may not be suitable for Target Observability (Rose RealTime term for visual target debug monitoring).
- Lack of control of target remotely (e.g. being unable to run developed application remotely from the toolset)
- Defects in Rose RealTime, Windows CE, Windows 2000 or eMbedded Visual C++
- Limitations in functionality in Rose RealTime, Windows CE, Windows 2000 or eMbedded Visual C++
- Development effort required to code within the toolset exceeds the effort required for manual coding

3.0 Thesis Report Outline

The thesis report will be given in the following format:

1. Introduction
2. Technology Overview
   - 2.1. Rational Rose RealTime
   - 2.2. Connexis
   - 2.3. Windows CE 3.0
   - 2.4. Compaq iPAQ Pocket PC
   - 2.5. Infrared Communications
3. Problem Description
4. Development Road Map
5. Target Port to Pocket PC
   - 5.1. Testing the Tool Chain
     - 5.1.1. eMbedded Visual C++ 3.0 Pocket PC "Hello World"
     - 5.1.2. eMbedded Visual C++ 3.0 Pocket PC "Infrared Hello World"
     - 5.1.3. Rose RealTime Pocket PC "Hello World"
     - 5.1.4. Connexis Pocket PC "Hello World"
   - 5.2. Tool Chain Assessment and Configuration
6. Development
   - 6.1. Development Iteration 1 - PingPong Capsule (Pocket PC - single process)
   - 6.2. Development Iteration 2 - Ping Client, Pong Server (Pocket PC - distributed processes)
   - 6.3. Development Iteration 3 - Ping Client (Pocket PC), Pong Server (PC) across infrared.
7. Revised Development Road Map
8. Example Application using Revised Development Road Map
9. Conclusions
   - 9.1. Rose RealTime for Infrared
   - 9.2. Lessons Learned
10. Recommendations
11. Bibliography

4.0 Literature Review

The literature review is presented in two parts, a "Brief Review" and "Extended Review". The intention is that the Brief Review presents the review in the form that can be scanned and, if desired, addition information can be found in the Extended Review. Also reviewed are the Web Resources from which some of the literature has been taken, this provides a context to how the information has been delivered.

A general comment on the literature reviewed. Most of the literature was generated by a company/association promoting or documenting a product/standard/concept. As the success of the things being promoted is dependant on adoption by new users, the literature is well written, well thought out and accurate. The information though will be biased towards the positive aspects of the product/standard/concept. For the purpose of this project proposal though, the merits of the products and standards are not under question and therefore the literature is reviewed against usefulness with regard to the Thesis topic.

Reviewer's Background: I am a Senior Technical Support Engineer at Rational Software supporting Rose RealTime and other embedded development products. I have a broad software/hardware knowledge developed from 13 years industry experience mainly in test and integration. I have no serious development experience of Windows applications. My research interests are in applying technology to improve processes, particularly to improving software development processes with visual software utilities.

Rose RealTime Literature

A general comment of the literature associated with Rose RealTime. As most readers would be unfamiliar with the concepts around Rose RealTime, all the literature reviewed (including this project proposal itself) has information targeted at anyone from novice to expert. The authors have done this task well but novice readers should be aware that Rose RealTime, as a Development concept, would require the benefit of practical experience with the toolset to fully appreciate what is being relayed in the more expert topics. On the flip side, expert users will find the introductory concepts repeated.

In addition to this, the limitations of static text and images have an impact on the readability of the information. Rose RealTime is a visual modelling tool and the application development process is a visual experience. The visual development experience is not static, the developer builds a system by "drawing" it, building it up and seeing the system state behaviour visually in real-time at each step. Unless the reader has witnessed this first hand, matching up the textual description with the events can be deceptively difficult.

Rational Rose RealTime - Release v2001a.04.00 Product Documentation (2001)

Brief: The product documentation contains several hundred pages of user information on different components of Rose RealTime. The information is the complete guide to Rose RealTime and is essential to all users of Rose RealTime whether experienced or not.

Extended: The information is provided as on-line help and hardcopy. The product documentation defines the user specifications of Rose RealTime and therefore accurate and under constant review (by users daily). The nature of the documentation being a grouping of different material (e.g. Connexis, Modelling Guide, Target Porting Guide, etc.) means that the style can vary from pure reference material to step-by-step tutorials. The documentation is good at all user levels, providing animated images for new users and detailed information where needed for experienced users.

The on-line version contains example models demonstrating various modelling techniques. Of particular interest to this project is the Socket Interface Model, which describes how to maintain message processing with threads that block (due to sockets). Also of interest are the Connexis "Ping Pong" models and associated tutorial, which will be the base test model for the Connexis part of this project.

Developing Real-Time Software With Rational Rose RealTime - v2001.03.00 Training Documentation (2000)

Brief: This literature is only of value in combination with the training course. Combined with the training course this provides valuable information and experiences for a new user.

Extended: As this is training documentation the information is summarised and expanded upon during the training (through the lecture notes). The literature is therefore not useful for a reference point of view. However, anyone having done the course and exercises may find it useful as a "memory jogger". The course provides a starting point for new users, taking them through Object-Oriented Analysis and Design through to creating a Telephone Call Model example. As stated in the general comment, experience is valuable in learning Rose RealTime and the course provides practical experience.

White Paper: Designing Software using a UML Case Tool that Supports Software Distribution. Terrence Barrington, Gustave Lamperez. Lucent Technologies, Advanced Optical Networking Center (2000)
http://www.rational.com/media/whitepapers/RoseRT_Case_Study_Lucent.pdf

Brief: A good short 5 page summary of using Rose RealTime and Connexis from a non-Rational Software users point of view. The information is based on an old

release of Rose RealTime so some of the information is no longer valid but it is still useful.

Extended: Lucent has taken time to assess Rose RealTime's good and bad points. The paper provides valuable information as it gives real life user experiences and lessons learned. Unfortunately the "publicly" available whitepaper does not contain the complete picture and references it gives to Lucent internal documents are not accessible.

Designing for Concurrency and Distribution with Rational Rose RealTime. Garth Gullekson. Rational Software White Paper (2000). http://www.rational.com/media/whitepapers/concurrencyroserealtime.pdf

Brief: Good information for someone unfamiliar with Rose RealTime and the concepts of using UML for embedded systems. People familiar with the concepts will also find useful information in the form a capsule design advice.

Extended: Garth Gullekson has many years experience with real-time systems and also the toolset. This paper is a polished article which takes the reader through the problems, solutions that active capsules and other real-time UML extensions provide, how Rose RealTime operates and how to approach design. This has good coverage of the problem domain, which gives depth to the explanation of the solutions. Also a good white paper to read for anyone who does not have access to Rose RealTime as it does not concentrate on the toolset functionality.

Windows CE 3.0 Literature

Windows CE 3.0 Application Programming. Nick Grattan, Marshall Brian. Prentice Hall (2001)

Brief: A good source for most elements of Windows CE programming excluding the user interface. This book also includes a CD which has Microsoft eMbedded Visual C++ 3.0 and working example code to go with the book.

Extended: Infrared communication is approached in this book only to a small extent but enough information is provided to be able develop a simple infrared application. There is also infrared example code provided which displays a list of infrared devices detected through the infrared port.

Although this book lacks information on infrared, it provides a good ground for the other information that would be needed for developing a Windows CE 3.0 application such as creating processes and handling files. Having eMbedded Visual C++ 3.0 and example code also enables the reader to correlate the books information with working examples. The infrared examples do work and demonstrate a simple application quite effectively without the need for a remote application to communicate with.

Microsoft eMbedded Visual C++ 3.0 - 3.00.0099.0 Microsoft Product Documentation (2000)

Brief: Comprehensive and well set out information on developing Windows CE applications using eMbedded Visual C++. Provides detail on infrared and has client/server example code.

Extended: Especially considering that eMbedded Visual C++ is provided free, the quality of the documentation is impressive. The documentation provides information on all areas of application development on Windows CE 3.0 and does so clearly.

Infrared communication is explained and example code given. The documentation is searchable and information about infrared applications can be found in several areas. The documentation mainly focuses in WinSock for infrared communications. It does mention raw IR, lower level control, but unfortunately does not provide much information about how to use it.

In addition to the basic information, the documentation puts infrared communications in perspective with the International Organization for Standardization Open Systems Interconnection (ISO/OSI) model. The documentation provides diagrams, which show simply and clearly the Windows CE communications services against the 7 OSI layers.

Get Your Windows CE Device Talking With IrDA. Michael Heydt. Microsoft Internet Developer Article (1999)
http://www.microsoft.com/mind/0599/wince/wince.htm

Brief: This article is now dated and many of the issues documented are now fixed with the current releases of the OS's. I would not recommend reading this but I would recommend locating it and keeping it for reference as it still has information that may be required for troubleshooting.

Extended: This was written against Windows CE 2.0 and Windows 2000 beta 2. The author explains the difficulties encountered due to lack of IrDA support. IrDA support is provided with Windows 2000 and these difficulties should not be encountered. The article should be noted though for the coverage of practical issues that would be useful for troubleshooting. There is information on installing IrDA drivers, verifying installation and a client/server application explained in some detail.

Compaq iPAQ Pocket PC Literature

iPAQ H3000 Pocket PC Reference Guide, Second Edition - Compaq Product Documentation (2000).

Brief: Basic user manual for Pocket PC owners. This is required for knowing the capabilities of the Pocket PC model.

Extended: An 116 page document, which provides basic user information. The manual is provided in the form of a PDF with the Pocket PC CD. Included is a section called "Getting Connected" which gives detail at a user level on infrared communication. The document also provides the specifications for the Pocket PC.

How Pocket PC "Talks" with a Cell Phone. Arne Hess. Club PocketPC Article (2000)
http://www.microsoft.com/mobile/pocketpc/columns/ppcomm.asp

Brief: Only contains a few paragraphs on Infrared. Of value only in the fact that it indicates a practical use and states that it is the "most convenient way to connect".

Extended: Although the article was sparse on detail it did influence the decision on the use of infrared for this project. Initially many communication mediums were being looked at but as the article pointed out, "all Pocket PCs have an infrared eye" and other mechanisms required additional hardware (cables or cards). So by using infrared any application developed could be used on all Pocket PCs.

Infrared Literature

A general comment on Infrared Data Association specifications. The specifications include state diagrams and state charts, which are ideal for Rose RealTime modelling. The reason that they are ideal is that the state machines can be defined within Rose RealTime exactly as they appear in the specification, code generated from them and runtime behaviour observed. However the ability to generate the code from the design and have the design (in this case state machines) exactly copy the specifications introduces some unique copyright issues. Fortunately the Infrared Data Association is approachable and will give consent as long as information from their specifications is acknowledged (this has been done for this project).

Also to note, is that the specifications contain a much higher level of detail than an application designer would require. The literature reviewed for Windows CE indicated that the WinSock services where used to access the infrared communications, this hides most of the detail given in the IrDA specifications. The "IrDA Infrared Communications: An Overview" by Megowan, Suvak and Knutson distils most of the information and is recommended as the best source, from the literature reviewed, of high level IrDA information for an application designer.

The Infrared Data Association has two main standards IrDA Data and IrDA Control. IrDA Data is of interest to this project as it deals with data transmission. IrDA Control is aimed at low throughput remote control devices (keyboards, tv controls, mice) and therefore not looked at as part of this project and review.

Technical Summary of "IrDA Data" and "IrDA CONTROL". Infrared Data Association. (2000) http://www.irda.org/standards/standards.asp

Brief: As the title suggest as summary of both IrDA Data and Control specifications. Nice 5 page description, which puts everything in context.

Extended: The amount of IrDA specifications makes this a very useful document. It is recommended that anyone tackling the IrDA protocols keep a copy of this to maintain a top-level view of where each specification fits in the overall plan.

IrDA Infrared Communications: An Overview. Patrick Megowan, David Suvak, Charles Knutson. (1998)
http://www.irda.org/use/pubs/Overview.PDF

Brief: This 21 page overview provides a good starting point for understanding the IrDA specifications. The information is also targeted at embedded systems.

Extended: The authors draw on experience not only from the Infrared Data Association but also their industry experience (Counterpoint Systems Foundry). Other overviews reference this paper and may even just present the same information without the Counterpoint name. The IrDA itself use this document as the sole document under their "Using IrDA" - "How it Works" website link. As there is effort required in understanding the IrDA protocol stack as a whole, this document is a must and should be read before any of the specifications.

Serial Infrared Physical Layer Specification, Version 1.3. Infrared Data Association (1998)

Brief: As the name of the specification suggests, this document describes the physical characteristics of an IrDA link. This information, although important, is probably not required from a programming perspective.

Extended: This document is written to describe to a hardware designer the physical specifications of a infrared link and communication nodes. From a programming perspective, there is information of value but it has to be searched for. Information contained which maybe of use are related to things which may impact the application usability e.g. spatial positioning of IR nodes. Although it has been stated that this is probably not required, it is still recommended as especially in embedded development, it is always important to understand the underlying hardware.

Serial Infrared Link Access Protocol (IrLAP), Version 1.1. Infrared Data Association (1996)

Brief: This document describes the Link Access Protocol, which is the layer above the Physical Layer. The information provided is important but is not easy to read. It is recommended that this document be skimmed if information is required.

Extended: Information in this specification is set out in an order that does not flow. Readers of this that do not have understanding of the specification will find this difficult to follow. It is recommended that the reader first understand the summarised version, (IrDA Infrared Communications: An Overview. Patrick Megowan, David Suvak, Charles Knutson), before attempting to read this. Once the reader understands the basics, the specification should be skimmed as it does contain information about performance of data links and other characteristics, which impact application development.

Link Management Protocol, Version 1.1. Infrared Data Association (1996)

Brief: This document describes the Link Management Protocol, which is the layer above the Link Access Protocol. Again the information provided is more than what an application designer would want but it is still recommended that this document be skimmed.

Extended: This is more readable than the Link Access Protocol but contains a lot of information, which would not be relevant to an application designer. There are important parts which should be noted but these are covered in the "IrDA Infrared Communications: An Overview". It is recommended that the document be skimmed and the structure and content be mentally noted for future reference. This document contains information about the flow control limitations, multiplexing of infrared links and Information Access Services, which should be looked at more carefully.

'Tiny TP': A Flow-Control Mechanism for use with IrLMP, Version 1.1. Infrared Data Association (1996)

Brief: This document describes how flow control is added to the Link Management Protocol. Again the information is provided in the "IrDA Infrared Communications: An Overview".

Extended: This is a short document (23 pages) but still has too much detail for an application designer. The recommendation is to note that this document exists for future reference.

'IrCOMM': Serial and Parallel Port Emulation over IR (Wire Replacement), Version 1.0. Infrared Data Association (1995)

Brief: IrComm is the specification to be only used for legacy applications. The specification itself recommends that IrComm should not be used.

Extended: This information would be of definite use for a legacy application but for this project this is not the case. The reasons for not using IrComm are due to mapping "wired" protocols to infrared. Due to the document not having relevance the overview was all that was read.

IrDA Object Exchange Protocol, IrOBEX, Version 1.2. Infrared Data Association (1999)

Brief: This protocol allows communication of arbitrary data objects between applications. From an application design point of view this contains valuable information and also structured to be readable for application designer.

Extended: It should be first stated the one of the goals of this project is to apply Connexis to infrared communications. If this is successful then the IrOBEX protocol would not be required as the protocols between applications would be defined in the Rose RealTime toolset. However, the IrOBEX protocol is still worth knowing about and this specification provides a lot of good information. There are not only specifications but also examples of how the protocol has been applied and test guidelines. The information would also be valuable as a complementary protocol to Connexis or for communication with no Rose RealTime applications.

Minimal IrDA Protocol Implementation (IrDA Lite), Version 1.0. Infrared Data Association (1996)

Brief: This is really an Appendix to the Serial Infrared Link Access Protocol and Link Management Protocol specifications. Without these, this document will not make sense (e.g. acronyms not defined). The specification gives ways to cut down on IrDA functionality but still maintain some compatibility with other IrDA devices.

Extended: This specification would only be of use if the reader, as an application designer, needed information on the minimal functionality an IrDA Lite device had. It states "This specification is intended to be a companion document to the IrDA IrLAP and IrLMP specifications", this is very true as it launches into descriptions of NDM and NRM without giving any indication to what NDM and NRM are. To be able to read this document the specifications mentioned will be required. This specification should be noted as existing but in most cases would not need to be referenced.

LAN Access Extensions for Link, Management Protocol, IrLAN, Version 1.0. Infrared Data Association (1997)

Brief: IrLAN is not supported on Windows CE and is designed for higher speed infrared connections. In the future when high-speed links for Pocket PCs may be standard this would provide useful information.

Extended: The IrLAN specification details "Ethernet style" infrared connections over high-speed 1.15 and 4 Mbps links. Unfortunately the Pocket PC is limited to 115 kbs and therefore this specification does not apply.

Programming With Infrared Sockets - Whitepaper, Prasanna .V, California Software Laboratories
http://www.cswl.com/whiteppr/white/infrared.html (1998)

Brief: A good whitepaper but the information is dated. It is worth noting that it is there as it does cover some things like "raw IR" in more detail than elsewhere. As far as programming Infrared, based on the content of this document, it appears that any information pre 1999 will be outdated.

Extended: Since this was written, infrared sockets are now accessible from Windows Sockets. With the new support for infrared, any previous methods should be avoided to maintain code portability to future releases. So indirectly, this whitepaper indicates the "use by" date of infrared programming literature with respect to Windows. The information is still valuable and should be noted, but for current programming practices it cannot be relied on.

Connecting Windows and Non-Windows Devices with IrDA. Mike Zintel http://www.irda.org/design/WindowsNonWindowsIrda.PDF (2000)

Brief: Very good information to have but there are accuracy issues. This article gives a run down on IrDA from an application programmer's perspective and is recommended to read.

Extended: Lots of useful information and easy to read. Unfortunately the information is inaccurate at some points, which brings into question the accuracy of the entire document. For example, it states "Information Advertising Service (IAS)" which is incorrect, IAS stands for "Information Access Service" as defined in the Link Management Protocol reference (in both this review and the article). Although not totally accurate, this article is well worth reading and will impart valuable information to an application designer.

Web Resources

Rational Rose RealTime (Rational)
http://www.rational.com/products/rosert/

This section of the Rational website provides an overview of the capabilities of Rose RealTime and also gives access (via the Support link) to "Tech Notes" which give detailed user information on a variety of topics. The information is intended for prospective customers (overview) and current customers that are looking for a solution to a question or usage problem (Tech Notes).

The Rational Edge (Rational)
www.therationaledge.com

This website is a Rational e-zine. This contains articles from Rose RealTime experts from within Rational describing aspects of RealTime development and using Rose RealTime for developing software. This information, although not directly useful for

IrDA development, does provide insight to the issues related to software development using Rose RealTime.

The Infrared Data Association
www.irda.org

This website is an essential resource for finding information on IrDA. The site contains IrDA specifications freely available for download. Also provided is information for developers and users, such as guidelines. According to their resume, the "IrDA is an International Organization that creates and promotes interoperable, low cost infrared data interconnection standards that support a walk-up, point-to-point user model." and the site content confirms that statement.

Pocket PC (Microsoft)
www.microsoft.com/mobile/pocketpc/

This section of the Microsoft website provides good information on the Pocket PC and has several articles on using IrDA and connectivity options. Though not too in-depth it still provides valuable information. The articles are well structured and give the background of the authors.

Windows Embedded Developer Center (Microsoft)
http://msdn.microsoft.com/embedded/

This provides on-line information in the form of the MSDN (Microsoft Developer Network) online library. This site has a very large pool of information and should also be considered an essential resource for in-depth Windows CE information.

Windows CE (Microsoft)
www.microsoft.com/windows/embedded/ce/

This section of the Microsoft website provides "infomercial" style information. Of limited value apart from the ability to download Visual embedded Tools (the development environment for Windows CE 3.0). The site is aimed at selling Windows CE rather than providing information about it.

IPAQ Pocket PC (Compaq)
www.compaq.com/products/handhelds/pocketpc/

This website provides basic information about the IPAQ Pocket PC. It also lists the compatible connectivity options and links the vendor's sites. Although it has limited content, it provides easy to get to and uncluttered information.

Windows CEcity
www.wincecity.com

This website is active location for Window CE and Pocket PC information. Lots of information available but has to be searched for. The discussion groups are also very active and informative.

5.0 Research Methodology

The research goals are to apply the Rose RealTime to a particular problem and, in the process of doing this, add to the knowledge how to apply Rose RealTime to a problem. The end result will be a specific problem solved (or reasons why it wasn t solved) and lessons learned, which could be specific to the problem or generic.

Action Research was chosen as the research methodology as applies an iterative cycle to the research process. The iterative nature of Action Research also lends itself to Rose RealTime, which promotes iterative development as the development methodology.

As stated in "An Overview of Common Research Approaches" (UTS IT Research Methods Study Guide, John Hughes) "A significant feature of action research is that it operates in cycles or involves a spiral process - action research proceeds by doing and by making mistakes in a self-reflective spiral of planning, acting, observing, reflecting, planning etc. This spiral is one in which feedback is going on in many ways at once and the research adapts to new influences and is modified and changed by events - the complex and creative business of real life can be accommodated."

In this the case of this project, Action Research provides the most effective means of extracting knowledge and making the research methodology adaptive enough to generate valid results. The methodology does this by both allowing a qualitative aspect (the researcher makes value judgements as a user) and allowing the process to adapt to a complex real world situation.

Action Research addresses the following real world complex problems:

Unknown behaviour – The researcher has no control over the internal workings of the items under research. Behaviour of the Operating System, Toolset, Compiler can only be found through practice. Action Research allows the behaviour to be discovered and actions taken based on what was discovered.

User perspective is required – The researcher needs to play a dual role of researcher and participant. During the process, the researcher needs to make qualitative decisions based on a users aspect. For example, infrared may be possible to use for development but there is no point in using it if there is some human aspect that makes it unusable a real world situation.

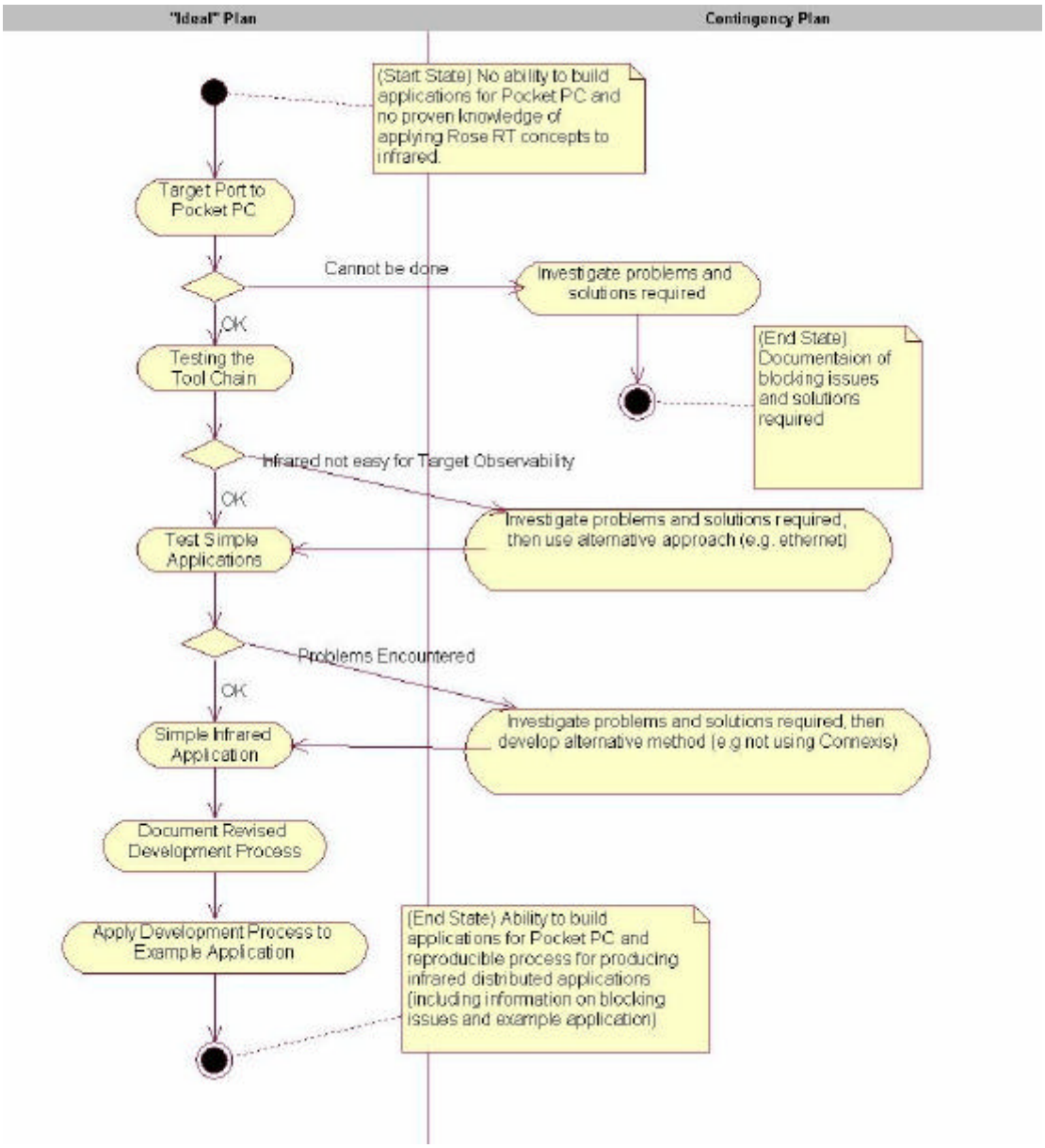
External influences – There are external influences, which may need to be reacted to. To maintain validity the research cannot exist in a "bubble", the results will need to apply to the current situation as it evolves. For example, it is expected that some or all of the products under investigation will change over the lifetime of the project as a result of bug fixes and upgrades. If these are not absorbed into the research, the final results could be invalid because they can't be applied to the current situation.

6.0 Project Timetable

The timetable for this project is unpredictable. It is expected that issues will be discovered which will slow or even completely block the development roadmap proposed. On the positive side, some issues may not appear or be solved by new releases of the various products over the duration of the project.

From the reviewing the literature, it is clear that there is a path that can be taken to achieve the goals of producing an infrared distributed application. From the known risks, it is also apparent that achieving the goal can be done within the timeframe even if blocking issues are discovered. The project timetable is therefore constructed in a flexible manner. The goal of this is to manage the time in a way that will allow the project to progress and be completed within one semester.

The time allocated to the project is 300 hours and the time used where appropriate at the time. The size of final application developed (if any) will then depend on the time left over from completing the previous activities. Below is a UML Activity Diagram, which represents the project plan based on "ideal" activities and "contingency" activities.


"Ideal" Plan
Contingency Plan
(Start State) No ability to build applications for Pocket PC and no proven knowledge of applying Rose RT concepts to infrared.
Target Port to Pocket PC
Cannot be done
Investigate problems and solutions required
(End State) Documentaion of blocking issues and solutions required
OK
Testing the Tool Chain
Infrared not easy for Target Observability
OK
Investigate problems and solutions required, then use alternative approach (e.g. ethernet)
Test Simple Applications
Problems Encountered
OK
Investigate problems and solutions required, then develop alternative method (e.g not using Connexis)
Simple Infrared Application
Document Revised Development Process
Apply Development Process to Example Application
(End State) Ability to build applications for Pocket PC and reproducible process for producing infrared distributed applications (including information on blocking issues and example application)