Abstract

UNU/IIST is a UN organisation located in Macau. Its mission is to transfer software technology to developing countries. The research and training programme of UNU/IIST centers on Fellowships in Macau and elsewhere, Research Projects, Advanced Development Projects, Curriculum Development Projects, Post-graduate Computing Science Courses, and Dissemination and Events. Serving developing countries in the Asia-Pacific region is part of the UNU/IIST mission. UNU/IIST looks forward to establishing joint training activity with The Regional Cooperation Office for City Informatization.
The programme of the City Informatization in the Asia-Pacific Region (CIAPR) aims to transform the digital divide into a digital opportunity in the Asia-Pacific region.

UNU/IIST is a UN organisation located in this region with a mission to help developing countries build up their capacities in the area of software technology, which is crucial to the development of their ICT. Serving developing countries in the Asia-Pacific region is part of the UNU/IIST mission. UNU/IIST is ready to contribute to the CIAPR programme.

This document briefs readers on the mission and recent activities of UNU/IIST. It provides UNU/IIST partners a basic knowledge about the institute.

1 Introduction

UNU/IIST (United Nations University International Institute for Software Technology) is a Research and Training Centre of the United Nations University (UNU). It is based in Macau, and was founded in 1991. It started operations in July 1992. UNU/IIST is jointly funded by the governments of Macau, People's Republic of China and Portugal with a US$30 million contribution to the UNU Endowment Fund. As well as providing two-thirds of the endowment fund, the Macau authorities also supply UNU/IIST with its office premises and furniture and subsidise fellow accommodation.

UNU/IIST is fundamentally concerned with assisting developing countries with the development and procurement of software technology. Although the growth of computer usage in such countries is high, most of them urgently need the establishment or expansion of both basic and advanced training, development and research facilities. Thus, there is both a lack of available software professionals for industrial growth and a chronic shortage of educators and trainers of the professionals of the future.

Moreover, most developing countries have few software companies and little experience in industrial software development. Local software development is essential in developing countries, not only to establish and strengthen local industry, but also to provide software in local languages and with cultural features adapted to the specific needs and conditions of each country.
Therefore, the mission of UNU/IIST is to assist developing countries in meeting needs and in strengthening capabilities in three areas:

- university education curriculum development,
- development of their software industry, and
- participation in international research.

## 2 UNU/IIST Programme

UNU/IIST’s programme centres on:

1. Fellowships in Macau and elsewhere
2. Research Projects
3. Advanced Development Projects
4. Curriculum Development Projects
5. Post-graduate Computing Science Courses
6. Dissemination and Events
7. Cooperation with Macau

### 2.1 Fellowships

UNU/IIST provides Post-graduate Training by conducting a Fellowship Programme.

UNU/IIST fellows must satisfy a number of criteria:

- they must have B.Sc. ---- with a M.Sc. being desirable
- they must be well-versed in the basics of software engineering and computer science
- they must have a relatively good level of English
Fellows are selected by UNU/IIST, and join in UNU/IIST projects to receive three types of training at UNU/IIST.

- Research training ---- whereby fellows are trained in doing research, in asking relevant scientific questions, and in the techniques for solving these. During their fellowship period, they are expected to write scientific reports and are encouraged to submit them to journals and scientific conferences.
- Development training ---- whereby fellows are trained in advanced development of software ---- in methods, principles, techniques and tools.
- Curriculum Development ---- whereby fellows spend typically 3~4 months at UNU/IIST, during which time they develop, together with UNU/IIST staff, course curricula and course material for post-graduate courses in their home countries.

Through its partner universities in developed countries, UNU/IIST also provides off-shore fellowships (ref. Sections 2.1.1 and 2.1.2). By the end of 2000, 216 fellows from 30 countries have received (or are receiving) training at UNU/IIST in Macau or abroad.

The following table shows the number of UNU/IIST fellows by country of origin and year of arrival. Among them a few are from developed countries who studied at UNU/IIST at their own expenses.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>China</td>
<td>10</td>
<td>4</td>
<td>5</td>
<td>9</td>
<td>11</td>
<td>11</td>
<td>13</td>
<td>21</td>
</tr>
<tr>
<td>Vietnam</td>
<td>2</td>
<td>3</td>
<td>8</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Indonesia</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Philippines</td>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>India</td>
<td></td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Mongolia</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Rep.of Korea</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poland</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ukraine</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Romania</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belarus</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Brazil</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Denmark</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Argentina</td>
<td></td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russia</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2.1.1 Offshore fellowships - Utrecht

As a continuation of UNU/IIST Research project "DesCaRTeS: Design Calculi and Research for Telecommunication Systems", UNU/IIST and the Faculty of Philosophy of Utrecht University are cooperating in providing research training at the postgraduate level in Utrecht to two fellows from Malawi and China. These two fellows have conducted research on specification and validation techniques in software development in telecommunications at Utrecht University since 1999. Fellow from Malawi had already completed her training and returned to Malawi in June 2000.

2.1.2 Offshore fellowships - Computer Science Department Development project

In 1998, UNU/IIST decided to extend the scope of its curriculum development activities by launching the project of Development of Computer Science Departments in Developing Countries, which aims to strengthen computer science teaching in universities in developing countries by training computer science teachers from these universities at partner universities in developed countries.

The lecturers from the developing countries generally spend one semester at one of the partner
universities, during which time they study several (generally four or five) courses offered by the partner university. These courses may be at either undergraduate or postgraduate level. For each of these courses, they receive from the partner university all the appropriate course material (lecture material, student's notes, course exercises, etc.), and UNU/IIST provides them with recommended text book(s) for the course.

When the lecturers return to their home university, they use the knowledge they have gained on the project, together with the course material and text books provided to them, as the basis for improving existing courses or introducing new courses into the teaching curriculum of their own university, thereby updating and expanding this curriculum.

In order to maximize the benefits from the project for any particular developing country, the project is run on a "knowledge sharing" basis ---- as far as possible the lecturers selected for the project from any given country study different sets of courses at overseas universities, and when they return home the knowledge and the course material and text books they acquire through the project are made available to other universities in the same country.

In each case, the partners are UNU/IIST, one partner university from a developed country, and one university from a developing country.

So far, three lecturers from Mongolia, four from Vietnam and one from Cameroon studied under the project at partner universities in Leicester, York, Belfast and Oxford in the UK, in Brisbane, Australia and Toronto in Canada. The project is now expanding to other developing countries and additional partner universities are actively being sought.

In addition to the activity described above, the project has also developed a recommended computer science curriculum for developing countries under a Special Service Arrangement with an experienced professor from Mongolia. This will be used in the future to help universities in developing countries set up new computer science departments, and will also be used to advise universities which already have computer science departments how best to develop and expand their teaching curriculum.

2.2 Research Projects
2.2.1 DeTfoRS: Design Technique for Real-Time Systems

The research at UNU/IIST aims at understanding foundational matters of more sophisticated software technology. UNU/IIST has chosen the area of Design Technique for Real-time, Reactive and Hybrid Systems as its main field of research, since this is one of the most active research areas of formal methods for software construction. Reactive Systems are those which maintain on-going interactions with their environments; Real-time Systems exhibit non-trivial and critical timing properties and are executed with resources that are limited in number and speed; Hybrid Systems include both continuous and discrete components.

Representative examples of Real-time, Reactive and Hybrid Systems are control systems where computers are used to decide control policies. Due to rapid development of processor and circuit technology, many modern systems, even consumer electronics, use software to control physical components. How are these systems to be analysed? How are they to be designed? How can they be verified? There are many possible approaches, and more questions than answers.

The objectives of the project are:

1. to train Fellows in research techniques,
2. to introduce Fellows to front-of-the-wave research topics and ensure that returning Fellows propagate the techniques to developing countries,

The major formal tool that UNU/IIST uses is Duration Calculus which is based on the Interval Temporal Logic. It was first investigated in the European ESPRIT project, ProCoS (Provably Correct Systems), by Zhou Chaochen and his colleagues at the beginning of the 1990s. Since 1992, Prof. Zhou has successfully built at UNU/IIST a strong group on Duration Calculus and its applications to Real-time Systems and Hybrid Systems. In the past three years, methodological issues have been studied along with representative case studies such as inverted pendulum, steam boiler, water tank and chemical concentration systems. So far, DeTfoRS has trained about 41 Fellows from 12 developing countries, and published about 38 papers in international journals and at conferences.

As DeTfoRS has trained many Fellows, and since some of the seconding institutions in
developing countries have taken up the DeTfoRS research agenda, UNU/IIST is now conducting off-shore research by supporting former fellows to organise research groups in their home countries to undertake research in the DeTfoRS area.

In 1997--2000, UNU/IIST has run three off-shore projects related to DeTfoRS:

1. The project "Model Checking for Real-time and Hybrid Systems in the Framework of DC" with the Department of Computer Science and Technology in Nanjing University.
2. The project "Semantics and Verification of Real Programs Using DC: Theory and Practice" with the Tata Institute of Fundamental Research, Bombay, India.
3. The project "Real-time System Modelling and Analysis Technique for Multimedia Systems" with the School of Computer Science and Technology, Beijing University of Posts and Telecommunications, China.

2.2.2 Hardware/Software Co-design

The design and analysis of an integrated and tightly coupled hardware/software system (e.g., a sensor and its associated software driver) is hard. As we are requiring more and more of our modern information processing systems and as the circuitry available on an integrated circuit has been growing rapidly every year, these systems are bound to be complex.

Conventional hardware description languages VHDL and VERILOG allow designers to freely mix different levels of design abstractions from the hierarchic, structural net list at its lowest level to the behavioural expression of design operation at its highest. However, these languages are not translated automatically to circuits. There are interactive synthesis tools to aid in the construction of clocked circuits from a subset of the languages. The products are then verified by simulation.

A program expressed in an imperative language is state-based; i.e., it can be interpreted as a relation between the initial values of program variables with their final values. After being compiled into machine code it can be executed at high speed by electronic circuitry. Existing formal methods support modelling, design and sound reasoning of software systems. However, most existing formalisms lack sufficient scope and applicability; they cannot take into account combinations of important characteristics such as reliability and security. At the software level
that is close to hardware, specification formalisms and associated development tools have not been designed.

The key challenge for the correct design of mixed hardware/software is the search for a unifying framework for hardware and software co-design that enables us to keep up with the fast growth in the complexity of semiconductor devices and their associated software. In the context of verifiable design of both compiler and hardware levels, a unifying framework is needed to handle the interface between various components, for example, the various formal languages and their underlying mathematical frameworks used at different stages and different levels of system development. The formal methods used in software-intensive systems have highlighted the complexity of establishing conceptual links between formalisms intended for different components in a given system. It clearly shows the desirability of having a unifying framework.

We have produced a DC-based denotational semantics for the VERILOG Hardware Description Language and examined its algebraic properties. We also present an operational semantics for VERILOG. The semantics is encoded using the logic programming language Prolog in a literate programming style. It enables the semantics to be directly animated using a Prolog interpreter. Using this approach allows the exploration of sometimes subtle behaviour of parallel programs.

We enrich DC with a number of advanced features. The framework provides a unifying means for presenting the features of the event-based hardware description languages and state-based programming languages.

UNU/IIST is collaborating with South Bank University of London on the research project "Co-design". We plan to validate the algebraic laws of VERILOG using mechanical tools, and to develop a rewriting system to decompose a program into software and hardware components.

2.2.3 Other Research Projects

The Research Group also undertakes the following joint research projects:

1. The project "Integration of the event-based and the state-based design methods" with Oxford University Computing Laboratory since 1999.
2. The project "Component-based Calculus" with Leicester University of UK and the University of Macau.

2.3 Advanced Development Projects

Advanced development projects are one of the main activities of UNU/IIST. They usually involve one to three phases, each lasting for 9 to 12 months, involving one to four Fellows and one or more UNU/IIST staff. They generally focus on software support for infrastructures and aim

1. to train Fellows from the public and private sectors: universities, research institutes, business and industry
2. to contribute to research by trying also to understand the nature of infrastructures
3. to propagate Design Calculi-oriented (i.e. Formal) Methods for software development to universities, business and industry
4. to help develop advanced, initially public domain software in close cooperation with industry and business
5. to help bring software producing and/or relying industries, businesses and other institutions of developing countries at last on a par with those of industrialised countries
6. to disseminate results, including abilities and software, to developing countries

The advanced development projects of UNU/IIST use the well-known RAISE (Rigorous Approach to Industrial Software Engineering) method, which was developed by a European ESPRIT project. Chris George of UNU/IIST was one of the founders of the RAISE method.

Examples of advanced development projects are listed below.

2.3.1 Enterprise Modelling, Analysis and Implementation

The project is about the application of formal methods to enterprise engineering, for business organisations in general and manufacturing organisations in particular. We study formally-based
notations, methods and tools to analyse, design and re-design an enterprise as an engineering artifact. We emphasise the need for building representations of enterprises at different levels of abstraction and the need for formal underpinning for such representations to allow comparing them. The issues range from building enterprise models, relating models at different levels of abstraction, analysis of models at a given level, refinement from abstract to concrete models, composition of models to represent virtual organisations, implementation using current information technology etc.

By addressing such issues formally, we aim to provide a technical context to express and support newly emerged concepts like outsourcing and virtual organisations, business process engineering and re-engineering, recycling and product life-cycle, lean and just-in-time production, concepts which are increasingly adopted by manufacturing organisations worldwide to address challenges of global competition and unstable market conditions.

2.3.2 Airline Business Computing

This project studies software support for the day-to-day operation of the airline industry, including financing, scheduling and timetabling of flights, ticketing and reservations, maintenance and availability of aircraft, passenger and cargo services, air traffic control, etc.

In the first phase of the project, one fellow from Vietnam Airlines carried out a detailed high-level domain analysis of the whole of the airline business and investigated the requirements for an airline business "management" system which would not only support the management of the business but which would also allow the integration of diverse software already in use in different departments of an airline.

The second phase of the project focused on the design of a specific software system for flight effectiveness analysis, a process which an airline performs in order to assess the success or otherwise of its flight operations. The second phase involved two fellows from Vietnam Airlines and produced a specification of one of the two components of a flight effectiveness analysis system, namely which data on cost and revenue are needed in flight effectiveness analysis and how they are analysed to generate IATA standard reports on flight effectiveness.
The third phase of the project, which was carried out by one Fellow from Vietnam Airlines, focused on two more specific software systems. The first of these dealt with future demand forecasting, which is a technique an airline company uses as part of its business planning process to estimate future market demand. The second was a system supporting the administration of a frequent flyer program, which is a bonus scheme operated by an airline to try to ensure that existing customers develop some loyalty to the company, thereby ensuring future sales. For each of these components, a demonstrator software system was also constructed.

2.3.3 MultiScript: Multi-directional, Multi-lingual Text System

The project is studying the creation and presentation of documents in which more than one language is used, for example a dictionary from one language to another. Particular emphasis is being placed on support for languages which are not read and written in the standard European style (left to right and top to bottom), for example Japanese and Chinese (traditionally written vertically in columns with the columns proceeding from right to left), Mongolian (also traditionally written vertically in columns, but with the columns proceeding from left to right), and Arabic and Hebrew (written horizontally from right to left), and on how these can be combined in a single document while all still retaining their traditional directionality.

This sort of system is already important in a wide range of applications, and is becoming increasingly so with the increase in international electronic communications: libraries, universities, government departments (particularly in countries where many different languages are used, for instance India), commercial industry, hospitals and tourist information services, often need to store or make available information in more than one language, and international trade and business agreements often have to be written in the native languages of all parties.

Two fellows from the National University of Mongolia in Ulaanbaatar worked on the first two phases of the project and developed a specification of the functionality of a MultiScript software system. This is now being used as the foundation for the design and implementation of a prototype software system in the third phase of the project. This is being built using the Delphi system on top of Windows NT so as to enable a tie-in with the project's work on the international standard encoding scheme described below. Much of the underlying functionality of the system has already been completed, though work on the top-level interface for displaying and printing documents is
still in progress.

The project places particular emphasis on allowing languages and scripts with different reading and writing directions to be intermixed in the same document while all still retaining their traditional directionality. This is in direct contrast to much of the existing software support for multi-lingual documents which tends to be uni-directional, often imposing the standard left-to-right horizontal directionality of the Latin script on all text.

The project is also contributing, under the International Organisation for Standardisation's working group ISO/IEC JTC1 SC2 Working Group 2, to the definition of an international standard encoding system for traditional Mongolian script which forms part of the ISO/Unicode international standard, a coding system which covers the majority of the world's languages. A proposal for the standardisation of the coding for traditional Mongolian script was jointly prepared (UNU/IIST, Mongolia, and China) and was presented at a meeting of WG2 in London at the end of September 1998. This proposal has recently passed through the third and final stage of the international review process and has now been officially adopted and published as the official ISO/Unicode international standard encoding for traditional Mongolian script.

### 2.3.4 Object-Oriented Design Patterns

Object-oriented patterns represent abstractions of good solutions to recurring problems in object-oriented software design. This abstractness means that a given pattern can be used in many different applications, which makes it a valuable tool for constructing reusable software and for helping object modellers achieve more effective results. However, patterns are invariably described informally in the literature, generally using natural language together with some sort of graphical notation, which makes it very difficult to give any meaningful certification of software developed using them.

In this activity, we have developed a formal model of patterns which can form the basis for demonstrating that a particular design conforms to a given pattern, and we have used this model as a basis for formal specifications of most of the patterns in the GoF catalogue (Gamma, Helm, Johnson and Vlissides; Design Patterns - Elements of Reusable Object-Oriented Software; Addison-Wesley, 1995). We have also investigated extending the model to allow matching an
object-oriented design against multiple patterns simultaneously. In future work we plan to look at
variants of patterns and at the possibility of using the formal model as the basis for software tools
which could help designers apply patterns.

2.3.5 Port Management

UNU/IIST has worked with two fellows from the Electronics Corporation of India Ltd. (ECIL) on
the domain analysis and requirements capture of a system for integrated port management, which
covers:

- Vessel Traffic Management: monitoring and controlling ship movement between the sea,
anchorages and berths, piloting, ship information
- Container Traffic Control: yard planning, container tracking, inventory, billing
- Cargo Management: monitoring and controlling cargo movement and storage, import and
  export document handling, billing
- Financial Management: billing for ships and other customers, computerised accounting
- Executive Information: top level information for port management to aid in control and
  planning.

Effective integration of these systems is essential for port operations to be efficient (minimum
storage times and movements of cargoes, containers and ships), for bills and other documents to
be correct, and for managers to be able to control and plan effectively.

2.3.6 Methods and Tools for Building Software from Components

Most software today is built from pre-existing components, often given in a form which allows
them to be used but not analysed (typically off-the-shelf components distributed as binary files).
Software design means composition and correctness of the design relies on the correctness of
individual components. This practice saves on time and the cost of development and results in
products which are easier to upgrade and maintain. But it also creates a host of basic but
technically challenging questions and problems: What in essence is a reusable component? What is the semantics of a component? How can we abstract away from the details of the semantics in order to decide suitability of a component for the design? What is the "best" abstraction level for this? Given two descriptions of one component (a specification and an implementation) how can we make sure they are consistent? How to check consistency if the implementation details are unavailable? To what extent can we adapt a component which fits only partly into our design? And so on.

The goal of the project is to study formally-based methods for software design from pre-existing components, including but not limited to the issues above. In particular, we look for a common ground to use together formal methods (to predict the result of putting components together) and fault-tolerance (to detect and recover from errors of individual components at run-time). Based on the results of such study we want to implement prototype tools to perform specialised tasks like e.g. calculating the semantics of composition, generating proof conditions, generating component wrappers to monitor their behaviour at run-time, discovering automatically the presence of redundancy among components, carrying out transformations to exploit this redundancy, etc. We plan to implement such tools rigorously using RAISE and integrate them at the end into a single environment: Composition Workbench.

2.3.7 Timed RAISE

RAISE allows time to be modelled but has no specific built-in facilities for dealing with real-time problems. Duration Calculus (DC) is designed to describe and analyse timing properties of hardware or software but does not have the features to support the description of data structures, algorithms or the modularity of medium or large-scale systems.

This project aims to integrate RAISE and DC. It also therefore involves close collaboration between the advanced development and research groups at UNU/IIST.

There are several aspects of the project:

Extension of RSL:

The features for describing time need to be added to the RAISE Specification Language
(RSL). Currently these are a type for representing time (the non-negative real numbers) and a \texttt{wait} construct for specifying delays. The static and dynamic semantics of these extensions need to be defined. The extended language is called Timed RSL (TRSL).

Integration of TRSL and DC:

- If DC formulae and TRSL descriptions can both be used in a specification of a system then there needs to be some defined relation between them. One possibility is to define DC in terms of RSL. This is possible but makes it complicated to reason about DC formulae. A second possibility is to define a semantics of TRSL in DC. This needs only to be a partial semantics that expresses the timing characteristics of a TRSL description, as the untimed characteristics can use the existing proof theory of RSL. This is the approach currently being investigated.

Combined method:

- The development methods of RSL and DC need to be combined to support the development of TRSL descriptions. The convenience and effectiveness of the method ultimately justifies the choices made in the previous two activities.

\subsection*{2.3.8 RAISE Tools}

The current RAISE tools are comprehensive, robust and suitable for industrial projects. They are free of charge for teaching and research. But they currently only run on Sun workstations, which are often not available in developing countries.

This project aims to provide tools for RAISE that are easily portable, including to PC environments, and freely available as source as well in executable form.

A second aim of the project is to involve universities in developing countries in developing extensions to the tools.

The project started at the end of 1997 with an RSL type checker. To this have been added so far a "confidence condition" generator, a pretty-printer, translators from RSL to SML and C++, and a construct to support animation of RSL specifications.
2.4 Curriculum Development Projects

UNU/IIST is trying to increase its impact by assisting universities in developing countries to develop their ability to teach the subjects in UNU/IIST's "agenda", specifically the use of design calculi in software development. This is a common feature of computer science curricula in developed countries but much rarer in the developing ones.

Fellows are professors or lecturers who come to UNU/IIST for 3~4 months to develop material for use in undergraduate and post-graduate courses. The material generally complements UNU/IIST's own material used in its advanced development courses. At the end of their stay, fellows take home the course material and software for the support of the methods being taught.

Fellows so far have come from institutions in India, China, Gabon, Argentina, Vietnam, Mongolia, Cameroon, Japan, DPR Korea and Senegal.

2.5 Post-graduate Computing Science Courses

UNU/IIST has developed a course on the RAISE Method in connection with the Advanced Development Projects, a course on the Duration Calculus in connection with the DeTfoRS project, a course on SDL and Process Algebra in connection with the DesCaRTeS project, and more recently courses on Algorithmics and Software Project Management. Each course lasts for 1 to 2 weeks, and the Duration Calculi course is usually for 1 week (occasionally shortened to 3 days or extended to 2 weeks). Courses are sometimes combined and last subsequently 2-4 weeks in total. UNU/IIST has also given other courses according to the need of developing countries and the availability of its resources.

The objectives of these courses are to: (1) propagate awareness of UNU/IIST agenda: the most advanced software development techniques, (2) provide training in software development techniques and other computing subjects based on the UNU/IIST resources, (3) identify, with leading staff of the host institutions, possible joint advanced development and/or research project
actions, (4) identify potential Fellows.

The following is a list of about 96 courses that have been given (up to the end of 2000):

**RAISE:**

- **1992:** China and India
- **1993:** Thailand, Vietnam and DPR Korea
- **1994:** DPR Korea, The Philippines and Indonesia
- **1995:** Vietnam and Malaysia
- **1996:** Vietnam (twice), India, Argentina (twice), Mongolia, Brazil and Russia
- **1997:** Japan, China (twice), Gabon, Cameroon, India and Argentina
- **1998:** Argentina, Japan, DPR Korea, China (twice)
- **1999:** Egypt, Malaysia (twice), Azerbaijan, DPR Korea, Argentina, Nigeria, India, China
- **2000:** Vietnam, Tunisia, Senegal, Philippines, Tajikistan, Argentina (twice), Cuba, Botswana, Mongolia, Kazakhstan, Mozambique, Zimbabwe

**Duration Calculus:**

- **1992:** China and India
- **1993:** Thailand
- **1994:** The Philippines and Indonesia
- **1995:** Vietnam and Malaysia
- **1996:** India, Argentina and Brazil
- **1997:** Russia (twice), Ukraine, Bulgaria, Romania, India and Argentina
- **1998:** India, China (twice), Vietnam
- **1999:** China (twice), Nigeria, India, DPR Korea
- **2000:** Vietnam, Tunisia, Senegal, Argentina

**SDL:**
How United Nations University International Institute for Software Technology Serves Developing Countries?

- **1996**: Turkey, Romania, Ukraine and the Philippines
- **1997**: China, South Africa, Brazil, Pakistan and Indonesia

**Creating Home Pages:**

- **1999**: Azerbaijan (twice)

**Formal System Design:**

- **1999**: Argentina

**Specification, Design & Implementation of Relational Programming:**

- **2000**: Brazil

**Programming Logic:**

- **2000**: Singapore

**Algorithmics:**

- **2000**: Bangladesh

In addition to these courses, UNU/IIST staff present seminars in many countries. These vary in length and may be from an hour to 3 days about UNU/IIST in general or about UNU/IIST's research and development projects. Lists of such seminars are given in the annual reports.

### 2.6 Dissemination and Events

UNU/IIST sponsors a small number of computer science software engineering and technology conferences which are directly related to UNU/IIST's R&D areas and which are held in developing countries. Such sponsorship is usually in the form of financial support to enable
participants from developing countries to present papers which have been accepted by the conference.

UNU/IIST co-sponsored the following events in 2000, by providing financial support to cover conference expenses to allow young researchers to attend:

- IFIP WG2.3 Workshop (September 4-5) and School/Seminar ---- State-of-the-Art on Program Design Using Logic, (September 6-13) in Tandil, Argentina
- Asian Computing Conference (ASIAN 2000), Penang, Malaysia, December 2000

In August 2000, UNU/IIST also organized an international symposium on software industry in developing countries under the International Federation for Information Processing, World Computer Conference 2000 in Beijing. With this symposium, UNU/IIST facilitated developing countries to exchange experiences and lessons in establishing and promoting their software industries.

Even though many developing countries have increasing access to international literature, copying facilities, computers and Internet, such access is still, by and large, expensive, and many institutions and/or countries are still lagging behind. Therefore, there is still a need to provide easy and cheap access to literature (reports, papers) and free software.

UNU/IIST dissemination thus provides reasonable amounts of copies of literature and free (public domain) software to institutions in developing countries world-wide.

UNU/IIST provides regularly (each three months) a selection of updated information on international literature, free software and technical reports of UNU/IIST and partner institutions to targeted institutions with instructions on how to procure such publications/software. Recipients of the service are able to request a limited number of copies of journal articles and UNU/IIST publications.
2.7 Cooperation with China

Serving its host country is part of the obligations of UNU/IIST. Without the generous support from the host country, UNU/IIST could not have been established or developed to its current level of activity.

China, in particular Macau, has high expectations for UNU/IIST to promote software technology locally. UNU/IIST has conducted the following activities in its host country.

1. UNU/IIST has signed an agreement with Macau University. Under the agreement, UNU/IIST provides courses and library resources, joint supervision of B.Sc. and M.Sc. students and joint research. The Macau University reciprocates by making available library resources and course opportunities for UNU/IIST Fellows. In 2000, UNU/IIST gave two courses in Macau University.

2. The Office for the Development of Telecommunications and Information Technology (GDTTI) of the Macau SAR government and UNU/IIST have signed an agreement to jointly organise IT training courses for Macau professionals and conduct IT projects in Macau. On 17 August 2000, GDTTI and UNU/IIST jointly organised at UNU/IIST the seminar "XML and Its Impact for E-Commerce", and in 15-19 January 2001 a course on "Software Project Management".

3. A joint Fellowship Programme with Macau Foundation for Cooperation and Development has been established. It provides two UNU/IIST fellowships in 2000 to support UNU/IIST training for young scientists from Lusophone countries and China.


5. The "Information Network Centre of Macau" (INCoM) project, the collaboration agreement for which was signed on 30 May 1997 between Macau Foundation, the State Science and Technology Commission for China, and UNU/IIST. The project is to establish the Internet Information Centre about Macau and China, to be operated on hardware donated by China. UNU/IIST's role is to organise the team for the development of the centre. Macau Foundation will administrate the operation of the center. The centre may eventually become an IT centre of Macau to provide Macau with various information services.
This project covered the design and establishment of an on-line database on university study in China for overseas students. Furthermore, an Overseas Macanese Database System was put into operation in May 1999 with text in Chinese and Portuguese.

UNU/IIST was instrumental in the design of the Macau UNESCO Center's Web Site and CD-ROM which was launched in July 1999. UNU/IIST is developing an Internet library system for the Macau UNESCO Center. The Internet library now includes 276 full-text books and magazines, more than 500 publications in its online catalogue, and 39 local artists' works in its Art Gallery.

6. UNU/IIST, together with the China Internet Information Center and the Stone Rich Sight Information Technology Company of China, established the "Macau1999" Web site. UNU/IIST was responsible for providing technical support to maintain the site. The site received a significant number of visitors, peaking at 127,394 pageviews in a single day during the week leading up to the handover of Macau to the People's Republic of China. UNU/IIST has been invited to participate in the continuation of this project to develop the site into a portal web site for Macau.

7. UNU/IIST has completed a project with Transmac, a local bus company, to develop jointly a Management Information System with respect to the application of IC card in its business. The Transmac Management Information System has been in operation since 20 March 1999.

8. Cooperating with the Macau Electronic Data Interchange Working Group, UNU/IIST has completed a feasibility study of development of the Macau Customs and Excise Computing System.

9. In order to further develop UNU/IIST training and research activities in the mainland of China, in 2000 UNU/IIST signed 7 MoUs with Chinese institutions. Among them 6 MoUs were with the institutions in the western part of China.

3 Conclusion

Two key factors for the successful development of information industries are the training of highly skillful working forces and take-up of the technology by public and private sectors. The primary need to promote the CIAPR programme is also skilled IT professionals, educators and
trainers. The shortages of skilled people are particularly acute in developing countries in this region. As a research and training institute on software technology, UNU/IIST considers it both a privilege and a responsibility to be part of such an endeavor to alleviate and finally resolve this human resource shortage problem.

More UNU/IIST information is available at:
http://www.iist.unu.edu

UNU/IIST research reports are available at:
http://www.iist.unu.edu/home/UNU/IIST/newrh/III/1/page.html

You can also request them by email
library@iist.unu.edu
(do not forget to mention your postal address)

by mail:
UNU/IIST
P.O. Box 3058, Macau
China