

POSTERS

POSTERS

Chair: Peri Tarr,
IBM T. J. Watson Research Center

Poster Session: Monday, 5:30 pm - 7:30 pm, Tampa Convention Center
Poster Display: Tuesday-Friday, Exhibit Hall

The poster session is an informal and highly interactive session that gives OOPSLA attendees the opportunity to engage one another in discussions about relevant, ongoing work and critical issues in key areas, and also to learn about work in areas with which they wish to become familiar. Researchers and practitioners describe their work in progress or to elaborate on work presented in other conference forums, while all OOPSLA participants can obtain rapid, low-cost introductions to interesting work and technologies in object-oriented software engineering and provide input and feedback directly to the authors. Posters are displayed throughout the conference, so they also provide attendees with the opportunity to learn about interesting work at their leisure, and they enable interactions to occur throughout the conference. Posters cover the same interest areas as the Technical Papers and Practitioner Reports.

The OOPSLA 2001 Posters program begins with a special session at the Welcome Reception on Monday evening, where all posters will be on display and the authors will be present to meet with attendees and discuss their work. For the remainder of the conference, the posters will be displayed in the Exhibit Hall, so that OOPSLA participants can view them at their convenience, and poster authors will be available as time permits. The goal is to encourage small groups of individuals interested in a technical area to gather and interact, and the poster session will be organized to facilitate such interactions, and to assist attendees in finding information about topics that interest them.

1 Domain-Specific Visual Languages

Juha-Pekka Tolvanen, *Jyväskylä University*
Jeff Gray, *Vanderbilt University/ISIS*
Steven Kelly, *MetaCase Consulting*
Kalle Lyytinen, *Case Western Reserve University*

An upward shift in abstraction leads to a corresponding increase in productivity. In the past, this has occurred when programming languages have evolved towards a higher level of abstraction. Today, domain-specific visual languages provide a viable solution for continuing to raise the level of abstraction beyond coding.

2 A Framework for Performance Monitoring and Modeling of Enterprise Java™ Beans Applications

Adrian Mos, *Dublin City University*
John Murphy, *Dublin City University*

We present a methodology that helps developers and system integrators understand and potentially correct the performance issues of an EJB-based system at an object-oriented level. Using this methodology, they will also be able to predict the behavior of their system when different user loads are applied.

POSTERS

3 Flow- and Context-insensitive Points-to Analyses for Java™: Extensions and Evaluation

Donglin Liang, *Georgia Institute of Technology*
Maikel Pennings, *Georgia Institute of Technology*
Mary Jean Harrold, *Georgia Institute of Technology*

This poster presents extensions to Steensgaard's and Andersen's algorithms to handle Java™ features. The poster also presents the empirical studies that evaluate the effectiveness of handling Java™ features using alternative approaches and the impact of the points-to information provided by these two algorithms on client analyses that use the information.

4 Why Java™ is not Suitable for Object-Oriented Frameworks

Dragos Manolescu, *Applied Reasoning Systems Corporation*
and the *University of Kansas*
Adrian Kunzle, *Skillgames*

Many business applications involve Java™ and object-oriented frameworks. Several characteristics of Java™ conflict with some key features of frameworks. These conflicts force the creation of "work-arounds" by developers. We show several examples that illustrate the tensions that exist between Java™ and object-oriented frameworks, and discuss how we solved them.

5 Workshop on UML Profile for Enterprise Distributed Object Computing

Fred Cummins, *EDS bluesphere*
Arne Berre, *SINTEF Distributed Information Systems*
Cory Casanave, *Data Access Technologies*
Hiroshi Miyazaki, *Fujitsu*
Yoshi Nagase, *Technologic Arts Inc./Consortium for Business Object Promotion*
David Zenie, *Iona Technologies*
Sandy Tyndale-Biscoe, *Open-IT, Ltd.*

OMG has a tentative specification for the modeling of recursive components and processes for distributed computing. The purpose of this workshop is to discuss how this UML profile might be refined, extended and incorporated in development tools, as well as to discuss related work that could leverage this profile.

6 Architectural Patterns for Usability

Len Bass, *Software Engineering Institute/Carnegie Mellon University*
Bonnie E. John, *Software Engineering Institute/Carnegie Mellon University*

Facets of usability that require architectural support such as cancellation, undo, and progress bars are identified. For each facet, an architectural pattern is described that supports the achievement of the facet. Facets of usability that require architectural support are difficult to add after completing the initial design of a system and, hence, it is critical to identify these facets prior to initial system design.

7 Must Java™ Development Be So Slow?

Albrecht Wöß, *J. Kepler University*

Starting a new VM each time an application is executed forces numerous classes to be loaded multiple times. This overhead significantly slows down the development of Java™ software. So don't do it! We plan to eliminate these bottlenecks with an open Java™ environment where only one VM hosts all applications and classes are loaded only once.

POSTERS

8 Results of the Educators' Symposium: Looking for Abstractions in the Real World

Rebecca Wirfs-Brock, *Wirfs-Brock Associates*

Alan McKean, *Wirfs-Brock Associates*

Jutta Eckstein, *Objects in Action*

This poster presents some results of the Educators' Symposium activity session on object-oriented modeling: Looking for Abstractions in a Concrete World: Candidates, Responsibilities, and Collaborations. CRC has traditionally stood for "Classes-Responsibilities-Collaborators." But classes are too concrete. Jumping to classes too soon results in a lifeless design.

9 Specification and Verification of Component-Based Systems Workshop

Gary T. Leavens, *Iowa State University*

Dimitra Giannakopoulou, *NASA Ames Research Center*

Murali Sitaraman, *Clemson University*

This poster summarizes results from the workshop on Specification and Verification of Component-Based Systems. The workshop's goal is to explore foundations for applying formal methods to component-based systems. The results and future work sections of the poster will be prepared during the workshop.

10 Optimization of Planar Gradient Coil Systems for a Mobile Magnetic Resonance Device by Genetic Algorithms Using Object-Oriented Design Techniques

Hartmut Popella, *RWTH Aachen*

Gerhard Henneberger, *RWTH Aachen*

This poster presents an object-oriented software tool for the optimization of a planar surface gradient coil system for magnetic resonance imaging (MRI) performed by genetic algorithms. The application of UML notation is well suited to the development of a numerical computation tool for the construction of gradient coil systems.

11 Objects, XML and Databases

Akmal Chaudhri, *IBM Informix Labs*

Awais Rashid, *Lancaster University*

The OOPSLA 2001 Workshop on Objects, XML and Databases will explore various issues in the integration and effective use of these technologies. This poster summarizes and highlights solutions to a set of issues that will be extracted from the workshop presentations and discussions.

12 Performance Monitoring of Large Global Distributed Systems

Doris Ressmann, *De Montfort University*

Amelia Platt, *De Montfort University*

Steve Rumsby, *De Montfort University*

Technological advances, particularly in object-based software design and communication networks, have laid the foundations for building more sophisticated and flexible distributed systems. Distributed execution of applications demands that the performance of executing applications be monitored. This poster presents two architectures that support performance monitoring.

POSTERS

13 MultiJava: Open Classes and Multiple Dispatch for Java™

Curtis Clifton, *Iowa State University*
Gary Leavens, *Iowa State University*
Craig Chambers, *University of Washington*
Todd Millstein, *University of Washington*

MultiJava is a backward-compatible extension to Java™ that includes open classes and symmetric multiple dispatch. It is the first full-scale programming language to support these features while retaining modular static type checking and compilation. The implementation of mjc, a MultiJava compiler, validates the language design.

14 Automation of Component Communication in Java™

Alexander Sakharov, *Verizon*

This poster presents a generic solution to the problem of automating the implementation of component communication in Java™. It applies to J2EE components as well. The users specify component communications directly in Java™. A communication adapter is automatically generated from the specifications. This solution takes advantage of component hierarchies.

15 MOO: A Programming Environment that Promotes Feelings of “Being There” or Presence

John Towell, *Carroll College*
Elizabeth Towell, *Carroll College*

A MOO is a networked, object-oriented, text-based virtual reality. Connected users control the behavior of avatars that interact with the environment. Most people who experience this environment report feelings of “being there” or presence, which has been utilized as a pedagogical tool to teach object-oriented concepts.

16 Evaluation of the Runtime Performance of Control Flow Structures for Dynamic Dispatch in Java™

Olivier Zendra, *INRIA / McGill University*
Karel Driesen, *INRIA / McGill University*
Feng Qian, *McGill University*
Laurie Hendren, *McGill University*

We present an ongoing study of control flow structures in Java™. We run a number of benchmarks on various JVM and hardware platforms, to characterize the performance of these structures when used to simulate dynamic dispatch implementations. Both execution patterns and control structures have a surprisingly large impact on performance.

17 Interdependence of Software Evolution and Development Process Evolution in Agile Methodologies

Christian Wege, *University of Tuebingen*

A software development process can become more successful by analyzing the produced artifacts and feeding the results back into the process. This research will produce techniques for analyzing the artifacts, show how to interpret them and list a collection of experiences made in real world projects.

POSTERS

18 Supporting Distributed Extreme Programming

Frank Maurer, *University of Calgary*

Sebastien Martel, *University of Calgary*

Extreme programming is arguably improving the productivity of small, co-located software development teams. We introduce an approach that overcomes the XP constraint of co-location by introducing a process-support environment, called MILOS, which helps software development teams to maintain XP practices in a distributed setting.

19 The Eclipse Platform Plugin Architecture

Jeff McAffer, *Object Technology International*

The Open Source Eclipse Platform (<http://www.eclipse.org>) has been designed for building integrated development environments (IDEs). In order to focus the generic functionality on something specific, the Eclipse Platform is equipped with a mechanism for discovering, integrating, and running third-party plugins and to host their tool-specific UI.

20 Introducing Patterns (or Any New Idea) into Organizations

Linda Rising, *Independent consultant*

Mary Lynn Manns, *University of North Carolina at Asheville*

Alan O'Callaghan, *De Montfort University*

Many people who have tried to introduce patterns (or any new idea) into organizations have found it to be difficult. This poster presents a pattern language to capture solutions to these challenges. A thumbnail of each pattern will be displayed, as well as ways the patterns have been used.

21 Domain-Specific Pattern Languages

Michael Kircher, *Siemens AG*

Prashant Jain, *Siemens AG*

Kirthika Parameswaran, *Telecordia Technologies*

Douglas Schmidt, *University of California, Irvine*

Angelo Corsaro, *Washington University, St. Louis*

The goal of this poster is to capture the results of two workshops, namely “The Three-Tier Architecture Pattern Language” and “Towards Patterns and Pattern Languages for OO Distributed Real-time and Embedded Systems.” While the domains of the two workshops are quite different, they share a common theme of capturing patterns and pattern languages by identifying the underlying forces.

22 Pedagogies and Tools for Object Oriented Teaching and Learning

Jürgen Börstler, *Umea University*

Isabel Michiels, *Free University of Brussels*

Many OO teaching approaches focus on the control structures part of a specific OO language. Even “object first” approaches do sometimes confuse students by starting out with non-typical examples/exceptions from the rule. We summarize results from workshops on issues in teaching and learning object orientation held at OOPSLA and ECOOP.

POSTERS

23 An Object Oriented Approach for Developing Finite Element Solvers

Dirk van Riesen, *Aachen Institute of Technology (RWTH)*
Gerhard Henneberger, *Aachen Institute of Technology (RWTH)*

An object-oriented approach to finite element calculations in electromagnetic, structural dynamics and thermal applications is presented. Special attention is given to the easy implementation of new element types and new formulations. The object-oriented design makes it possible to link in external libraries for specific tasks, like MTL/ITL for the equation system.

24 Transmigration of Object Identity: The Programming Language Gilgul

Pascal Costanza, *University of Bonn*

Gilgul is an extension of Java™ that strictly separates the notions of reference and comparison that are traditionally subsumed in the concept of object identity. This allows for the introduction of new operations that open up new degrees of flexibility by providing a mechanism for unanticipated, dynamic software adaptation.

25 FCL Checker: Detecting Structural Errors in Framework Based Development

Daqing Hou, *University of Alberta*
H. James Hoover, *University of Alberta*

Object-oriented frameworks are hard to learn and use. The capability to automatically detect errors occurring at the boundary between frameworks and applications is crucial to mitigate the problem. We introduce the notion of framework constraints and the language FCL to formally specify and check them.

26 Transactions Meet MOM—System Support for Integrating Distributed Object Transactions and Messaging in Java™ and MQ Environments

Stefan Tai, *IBM Research*
Thomas Mikalsen, *IBM Research*
Isabelle Rouvellou, *IBM Research*
Stanley Sutton, *IBM Research*

The Dependency-Spheres project explores concepts and middleware system support for distributed transaction processing across object and messaging components. The objective is to enrich standard object middleware and messaging middleware to provide for an increased level of reliability for their use in combination in enterprise systems.

27 Reducing Proof Burden in Object-Oriented Verification

Francis Tang, *University of Edinburgh*

Formal verification of programs is generally accepted to be laborious and time consuming. The use of verification condition generators can significantly reduce the amount of work required. We demonstrate such an approach for object-oriented programs, using Euclid's algorithm as an example.

POSTERS

28 Java™ Component Development in Jiaazzi

Sean McDermid, *University of Utah*
Matthew Flatt, *University of Utah*
Wilson Hsieh, *University of Utah*

This poster shows how Java™ developers can use Jiaazzi to enhance their Java™ code with constructs for large-scale binary components. We show examples that use small source files and development flows to illustrate how to use Jiaazzi.

29 Multiple Executable-Filed Java™ Virtual Machine

Satoshi Numata, *Osaka Electro-Communication University*
Hirotaka Uoi, *Osaka Electro-Communication University*

Java™ programs contain safe parts, which are compiled into Java™ machine code, and unsafe parts, which are compiled into native machine code. We propose a new Java™ Virtual Machine composition, which comprises two executable files: one executes Java™ machine code, and the other executes native machine code. We believe this completely separates unsafe parts from safe parts, and it also improves safety and efficiency.

30 A Coordination Methodology and Technology for Agile Businesses

Luís Andrade, *ATX Software SA*
José Luiz Fiadeiro, *Univ. de Lisboa*
João Gouveia, *Oblog Software SA*
Georgios Koutsoukos, *Oblog Software SA*
Michel Wermelinger, *Univ. Nova de Lisboa*

Coordination contracts are a modeling primitive that facilitates the development of systems subject to frequently changing business requirements. This is achieved through non-intrusive superposition of adaptors and connectors, corresponding to volatile business rules, on the components that implement core, stable, services. The poster addresses both methodological and technological support.

31 Concern Space Modeling in Cosmos

Stanley Sutton Jr., *IBM Thomas J. Watson Research Center*
Isabelle Rouvellou, *IBM Thomas J. Watson Research Center*

Cosmos is a schema for modeling software concerns across the life cycle. It defines a metamodel, including concern types, relationships, and predicates, for modeling multidimensional concern spaces. Cosmos allows concerns to be modeled independently of development formalisms, tools, and methods, and it complements and supports advanced separation of concerns technologies.

32 STOOP: The Sable Toolkit for Object-Oriented Profiling

Rhodes Brown, *McGill University*
John Jorgensen, *McGill University*
Qin Wang, *McGill University*
Karel Driesen, *McGill University*
Laurie Hendren, *McGill University*
Clark Verbrugge, *McGill University*

Many tools are available to profile and visualize Java™ programs. However, those we have encountered are limited by the fixed set of data they collect. Frustrated by this limitation, we developed STOOP: the Sable Toolkit for Object-Oriented Profiling, a framework for creating tools to collect and visualize arbitrary profile data.

POSTERS

33 Handling Crosscutting Constraints in Domain-Specific Modeling

Jeff Gray, *Vanderbilt/ISIS*
Ted Bapty, *Vanderbilt/ISIS*
Sandeep Neema, *Vanderbilt/ISIS*
James Tuck, *Vanderbilt/ISIS*

Domain-specific models for embedded systems often contain constraints that aid in stipulating design criteria. These constraints, however, are typically scattered across a model hierarchy in such a manner that it is difficult to reason about the effect and purpose of each constraint. This poster describes an approach for providing better separation of concerns with respect to constraints.

34 Results of the Workshop: Human Issues on Agile Processes

Jens Coldewey, *Coldewey Consulting*
Jutta Eckstein, *Objects in Action*
Pete McBreen, *McBreen Consulting*
Alastair Handley, *Pragmatic Software Consulting Services Ltd.*

Many of the values stated by the “Manifesto for Agile Software Development” are Human Issues in Agile Processes. These issues affect team- as well as client management and team collaboration. The poster presents the results of the corresponding workshop, whose goal is to identify and illuminate these human issues.

35 Enterprise-Scale Java™ Object-To-Relational Persistence Strategies Compared

Changrong Ji, *The St. Paul Companies*
Casey Phipps, *The St. Paul Companies*
Alan Weiss, *The St. Paul Companies*
Wen Ji, *Independent Professional Services*

Enterprise-scale Java™ object-to-relational persistence strategies are briefly outlined and compared. Domain-integrated persistence frameworks are compared to distributed persistence services, and inheritance vs. metadata approaches are contrasted. We also include a discussion of how EJB, JDO and O/R mapping tools such as TOPLink fit into each strategy.

36 Beyond Design: Patterns (mis)used

Christa Schwanninger, *Siemens AG*
Elisa Baniassad, *University of British Columbia*
Vera Seidel, *Siemens AG*
Gail Murphy, *University of British Columbia*

This poster visualizes the results of the OOPSLA workshop called “Beyond Design: Patterns (mis)used”. This workshop is intended to collect attempts to use and perform research on aspects of current patterns beyond the act of transferring design experience. This could include research areas such as reverse engineering programming languages.

37 Model Driven Analysis & Design of XML Schemas

David Carlson, *Ontogenics Corp.*

The concept of model driven architecture (MDA) has become a focal point for the Object Management Group. This poster describes two approaches, using two different UML profiles, for refining abstract, platform-independent models (PIMs) into platform-specific models (PSMs) that can be used to generate XML Schemas.

POSTERS

38 Design Aspects for Describing Frameworks

Federico Balaguer, *University of Illinois at Urbana-Champaign*

This poster presents an extension to UML for describing design aspects of frameworks. Aspects are documented by applying a UML Profile called “Framework Description” to class diagrams. Design Aspects of frameworks are useful for reasoning about extension and instantiation scenarios, as well as for designing the applications that rely on them.

39 Visualizing Indirect Branch Hot Spots in Object-Oriented Programs

Matthew Holly, *McGill University*
Karel Driesen, *McGill University*

We demonstrate four visualizations of indirect branch instructions corresponding to switch statements and virtual function calls in object-oriented programs. Spatial and Temporal hot spot visualizations highlight code locality. Footprints show dynamic program size while prediction profiles visualize the regularity of a program phase.

40 An Object-Oriented Framework for Distributed Numerical Computations

Roxana Diaconescu, *Norwegian University of Science and Technology*

Writing concurrent distributed memory applications requires skills beyond the background of a researcher experimenting with scientific problems. We propose a component framework to capture the concurrency infrastructure for dynamically distributed numerical applications. We focus on the problem of concurrent Finite Element Method (FEM) solution of Partial Differential Equations (PDEs) for general (unstructured) meshes.

41 Autonomous Points in Component Composition

Vladimir Mencl, *Charles University, Prague*

Current technologies provide only primitive support for managing configuration of applications composed of software components. Advanced facilities are needed, namely, for specifying the points in a component hierarchy suitable for initiating an update or acquisition from an independent vendor. Autonomous points proposed in this poster reflect these requirements.

42 ArchJava: Connecting Software Architecture to Implementation

Jonathan Aldrich, *University of Washington*
Craig Chambers, *University of Washington*
David Notkin, *University of Washington*

ArchJava is a small extension of Java™ that integrates software architecture smoothly into Java™ code. Our approach enables reliable reasoning about system structure, because ArchJava ensures that the actual code corresponds to the conceptual architecture. This poster describes the ArchJava language, the downloadable tools, and our initial practical experience.

43 Comanche, Rapid Web Development for the Rest of Us

Bolet Kerimbaev, *Georgia Institute of Technology*

Our goal is to enable rapid development of web applications by non-professional programmers. Rapid development is achieved through a simplified model of building web applications, availability of familiar development tools, and automation of some tasks. Performance tuning ensures that dynamic applications retain good performance.

POSTERS

44 Design Support for Aspect-Oriented Software Development

Christina von Flach G. Chavez, *Pontifical Catholic University of Rio de Janeiro (PUC-Rio)*

Carlos J. P. de Lucena, *PUC-Rio*

We propose a design model for aspect-oriented software development, which incorporates the main features of aspect-oriented programming and is language- and process-independent. A set of design principles for aspect-oriented modeling is under investigation. A representative case study and a development scenario illustrate the suitability of the proposed ideas.

45 Concepts Reuse for Requirements Specifications

Walaa-Eldeen Mohamed Bakry, *Middlesex University*

Successful specification reuse in software development hinges on finding a mechanism that links new requirements to old ones. This research proposes Concepts Reuse as a new approach to specification reuse. It provides a classification scheme based on generic patterns that aid the matching process between old and new requirements.

46 OOSPICE – Bringing Objects and Components to Process Improvement and Capability Determination

Mark Woodman, *Middlesex University*

Freiderich Stallinger, *Kepler University Linz*

Alec Dorling, *University of Borås*

OOSPICE is an EU/Australian project to extend the SPICE approach to software process improvement and capability determination to cover object-oriented and component-based development. Major deliverables include: a unified CBD process metamodel, a tool-supported CBD assessment methodology, a CBD method and tool, and extensions to the ISO/IEC 15504 process assessment standard.

47 Separation of Concerns in Multi-Agent Software Engineering

Viviane Silva, *PUC-Rio*

Otavio Silva, *PUC-Rio*

Alessandro Garcia, *PUC-Rio*

Christina Chavez, *PUC-Rio*

Carlos Lucena, *PUC-Rio*

Agent technology has been revisited as a complementary approach to the object paradigm. Although objects and agents have many similarities, the introduction of agents in the object model poses new problems. In this context, we present our approach for dealing with the intricacies of developing agent systems using recent advances of separation of concerns techniques.

48 Raising the Level of Abstraction of Design Models

Joern Bettin, *Equinox Software Architects*

We have used a model-driven software development approach and template-based code generation techniques to build a complex Java™ application within the electricity industry. Our approach allows us to significantly raise the level of abstraction of design models, results in highly compact representations of application architecture and design, and highlights weaknesses of the UML and current UML tools.

POSTERS

49 A Reusable Design for Building Dynamically Programmable and Workflow-enabled Object-Oriented Software

Reza Razavi, *University of Paris 6 (LIP6)*

“Expert-programmable” software is software that is designed to allow non-programmer domain experts to define desired classes by defining at runtime new types of objects, their structure and behavior. There is currently no standardized design for creating such software. The author proposes a solution supporting dynamic composition of dynamically defined, workflow-enabled services.

50 EasyMock: Dynamic Generation of Mock Objects

Tammo Freese, *OFFIS*

This poster presents a Java™ library called EasyMock. EasyMock provides dynamically generated mock objects for interfaces by using Java™ proxies. This simplifies unit testing by allowing the generation and usage of mock objects directly within the test code.

51 Third Workshop on Best-Practices for the Design and Implementation of Business Rules

Ali Arsanjani, *IBM*

Joe Yoder, *Joe Yoder Enterprises*

Alan Abrams, *Cambridge University*

Reza Razavi, *Université Pierre et Marie Curie*

Jeff Oakes, *The Hartford*

Keith Levi, *Maharishi University of Management*

Businesses implement their information systems upon the pillars of business rules. Yet, writing scalable, robust, and adaptable business rules for business objects in today’s e-business and n-tier architectures is a challenge! This workshop will categorize and capture, in pattern format, best and successful practices in the design and implementation of business rules within the context of business objects and components.

52 Retest: A Regression Testing Tool for Java™ Software

Alessandro Orso, *Georgia Institute of Technology*

Donglin Liang, *Georgia Institute of Technology*

This poster presents a technique that effectively selects, from the test suite used to test the original version of a Java™ program or subsystem, all test cases that may reveal a fault in the modified version of the software. It also presents a system that implements the technique and empirical studies that demonstrate its effectiveness.

53 The Jalapeño Research Virtual Machine for Java™

Bowen Alpern, *IBM Research*
 Matthew Arnold, *IBM Research*
 C. R. Attanasio, *IBM Research*
 Maria Butrico, *IBM Research*
 Jong-Deok Choi, *IBM Research*
 Anthony Cocchi, *IBM Research*
 Julian Dolby, *IBM Research*
 Stephen Fink, *IBM Research*
 David Grove, *IBM Research*
 Michael Hind, *IBM Research*
 Mark Mergen, *IBM Research*
 Tom Ngao, *IBM Research*
 Igor Pechtchanski, *IBM Research*
 Vivek Sarkar, *IBM Research*
 Stephen E. Smith, *IBM Research*
 Peter Sweeney, *IBM Research*
 Martin Trapp, *IBM Research*

Building an infrastructure to explore Java™ design and implementation issues exceeds the resources of most academic research projects. This poster provides an overview of Jalapeño, a state-of-the-art research virtual machine for Java™ infrastructure, written in Java™ at IBM Research. Source for Jalapeño (available under academic license) has been acquired by at least a dozen universities.

54 The Aspect Mining Tool - Support for Concern Mining

Jan Hannemann, *University of British Columbia*
 Gregor Kiczales, *University of British Columbia*

Refactoring legacy systems to improve modularity is difficult since the current modularization may have left some concerns unextracted. To identify the complete concern code despite tangling and scattering, developers need the right information about the system. We propose combining lexical and structural queries to identify concern extents.

55 Demeter Aspects

Karl Lieberherr, *Northeastern University*
 David Lorenz, *Northeastern University*
 Doug Orleans, *Northeastern University*
 Johan Ovlinger, *Northeastern University*
 Mitchell Wand, *Northeastern University*
 Pengcheng Wu, *Northeastern University*

The Demeter group summarizes its activities in aspect-oriented software development (AOSD). Aspectual Collaborations are an extension of adaptive plug-and-play components with method replacement to modularize and parameterize aspects. Predicate Dispatch Extension adds around methods and other extensions to predicate dispatching to provide a flexible AOSD system.

POSTERS

56 Tool Support for Managing Crosscutting Concerns in Existing Artifacts

Elisa Baniassad, *University of British Columbia (UBC)*

Martin Robillard, *UBC*

Albert Lai, *UBC*

Gail Murphy, *UBC*

Software developers spend a lot of time handling crosscutting concerns. Tool support can help a developer find, understand, and manage these aspects. This poster describes three tools we have built to help address these needs for both source-level and design-level artifacts.

57 Separation of Distribution Concerns in Distributed Java™ Programming

Michiaki Tatsumori, *University of Tsukuba*

We propose an aspect-oriented distributed programming tool, with which programmers can specify the distribution aspect of program simply and separately from a non-distributed Java™ program, to enhance the modularity of a program. The aspect-weaver of this tool is a bytecode translator implemented as a customized class loader in Java™.

58 Assessing the Quality of Object-Oriented Designs

Ralf Reißing, *University of Stuttgart*

Design quality is vital for reducing software cost. Unfortunately, it is unclear what design quality really is. This work wants to clarify the notion of design quality and make it measurable. A quality model is created that can be used for design assessment, comparing design alternatives, and design improvement.

59 A Multithreaded Concurrent Generational Garbage Collector for Java

Chia-Tien Dan Lo, *Illinois Institute of Technology*

J. Morris Chang, *Illinois Institute of Technology*

A new multithreaded concurrent generational garbage collector based on mark-sweep with the assistance of reference counting is proposed and implemented. The scheme takes advantage of multiple processors in SMP systems and the merits of threads. Furthermore, it reduces garbage collection pauses and enhances garbage collection efficiency. Measurement results are studied.

60 On the Syllogistic Structure of Object-Oriented Programming

Derek Rayside, *University of Waterloo*

Kostas Kontogiannis, *University of Waterloo*

This poster demonstrates a prototype implementation of a novel program browser for understanding polymorphic calls in Java programs. The browser's design was first presented at ICSE 2001 and is inspired by Aristotle's syllogism. The prototype is built on IBM's new WebSphere Studio Workbench (aka Eclipse) technology.

61 Integration of Independently-Developed Object-Oriented Designs

Adam Batenin, *University of Bath*

Programming technology should facilitate the independent development of object-oriented designs and their subsequent composition to create complete programs. We show that even in an environment where a limited independence is granted, relatively simple programs are difficult to integrate using present technologies. We propose techniques to address these issues.

POSTERS

62 An OCL Query-Based Debugger for C++

Chanika Hobatr, *Clemson University*

Development using object technology can make debugging a daunting task due to the number of generated objects and the complexity of their interrelationships. We propose a debugging tool that permits the expression of complex relationships to be formulated easily and evaluated efficiently. OQBD is a query-based debugger for C++ programs using queries formulated in the Object Constraint Language.

63 Hyper/J™: Supporting Decomposition, Integration, and Evolution of Java™ Software

Harold Ossher, *IBM Thomas J. Watson Research Center*

Peri Tarr, *IBM Thomas J. Watson Research Center*

Vincent Kruskal, *IBM Thomas J. Watson Research Center*

Hyper/J™ supports a new approach to constructing, integrating and evolving standard Java™ software called Multidimensional Separation of Concerns. Developers can decompose and organize software according to multiple, arbitrary criteria (concerns) simultaneously—even after the software has been implemented—and integrate the pieces into larger-scale components and systems. Hyper/J works on standard Java class files and is freely available.