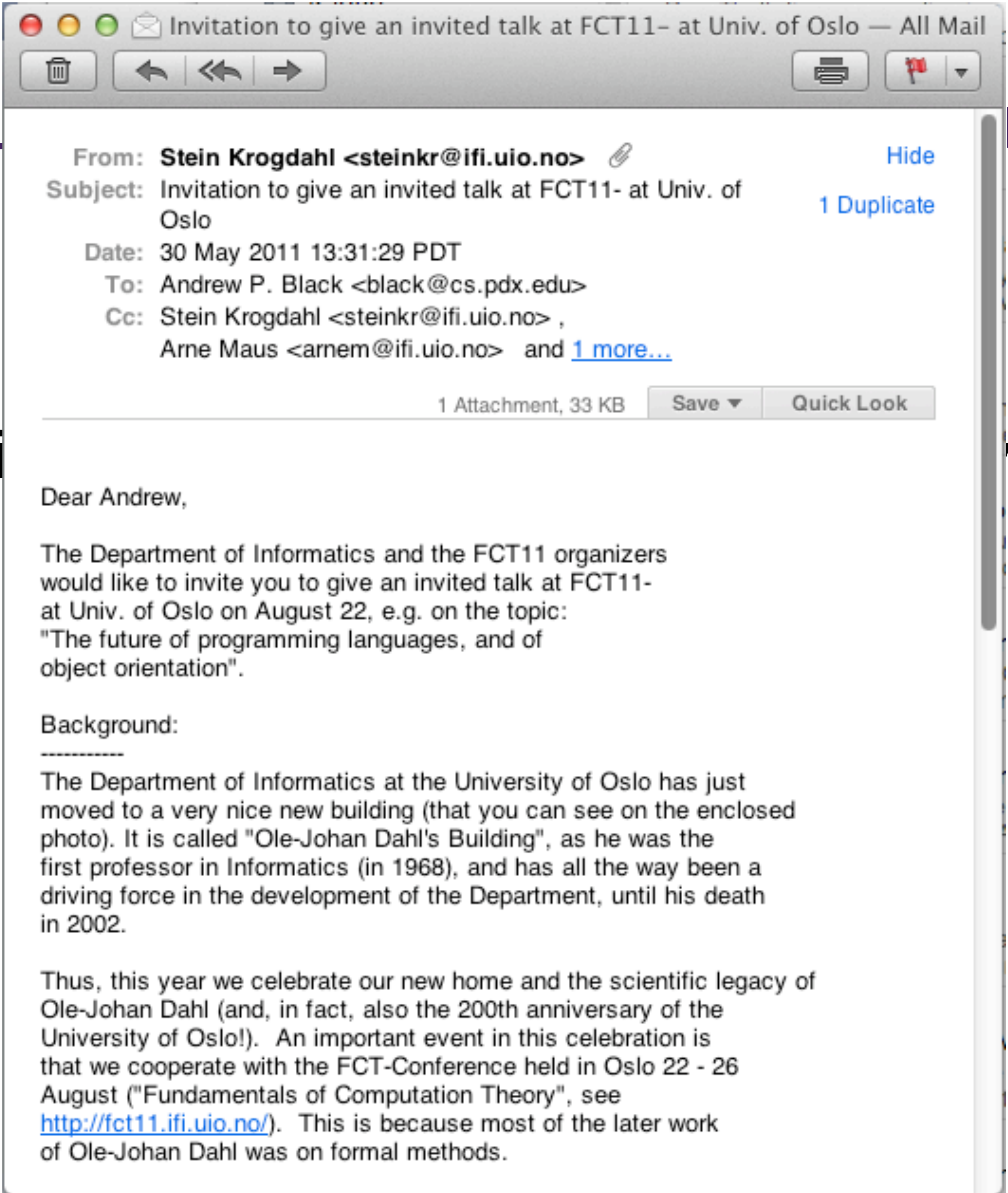
Journal Publishing

Case Study

Ole-J ming

- Invi 2011



Invitation to give an invited talk at FCT11- at Univ. of Oslo — All Mail

From: **Stein Krogdahl <steinkr@ifi.uio.no>**
Subject: Invitation to give an invited talk at FCT11- at Univ. of Oslo
Date: 30 May 2011 13:31:29 PDT
To: Andrew P. Black <black@cs.pdx.edu>
Cc: Stein Krogdahl <steinkr@ifi.uio.no> , Arne Maus <arnem@ifi.uio.no> and 1 more...

1 Attachment, 33 KB

Dear Andrew,

The Department of Informatics and the FCT11 organizers would like to invite you to give an invited talk at FCT11- at Univ. of Oslo on August 22, e.g. on the topic: "The future of programming languages, and of object orientation".

Background:

The Department of Informatics at the University of Oslo has just moved to a very nice new building (that you can see on the enclosed photo). It is called "Ole-Johan Dahl's Building", as he was the first professor in Informatics (in 1968), and has all the way been a driving force in the development of the Department, until his death in 2002.

Thus, this year we celebrate our new home and the scientific legacy of Ole-Johan Dahl (and, in fact, also the 200th anniversary of the University of Oslo!). An important event in this celebration is that we cooperate with the FCT-Conference held in Oslo 22 - 26 August ("Fundamentals of Computation Theory", see http://fct11.ifi.uio.no/). This is because most of the later work of Ole-Johan Dahl was on formal methods.

- Invitation to speak at FCT’11: 30 May 2011
- Accepted: 7 June 2011
- Talk delivered: 22 August 2011

FCT Post-Proceeding

- 26 August 2011

  Dear Andrew,
  … There is a chance that we will make a special issue of papers from FCT 2011 in the journal Information and Computation.

  Then, a paper based on your talk would be welcome, and we would (preliminary) like to hear if this would be of interested to you.

  It might be of interest. There would, however, be a lot of work required to turn this material into a coherent paper. When would the deadline be?

- 10 November 2011:

  Dear Andrew,
  I would like to inform you that there will be a special issue of selected papers from FCT 2011 for the Elsevier journal Information and Computation, and we would very much like to have a submission from you related to the work you presented in Oslo.

- 21 November 2011:

  Dear Olaf,
  I'm going to plan to write a paper over the Christmas break. What is the due date?

  That sounds very good. How would the date January 15 fit you?

- 19–21 January 2012:

Dear Olaf,
As you see, I have missed the deadline. I made some progress over the Christmas break, but am still not a quarter of the way through the paper that I wanted to write. The last two weeks have been the start of term here, and I have not been able to spend a minute on this paper.

Next week I'm traveling, and may get to do some writing. But I'm not going to be able to get near to finishing even a first draft.

What to do?

Dear Andrew,
Yes I understand. I could give you a month extension, i.e, Feb 15; would that be helpful? We could even try end of Feb., but it is hard to give more than that, due to the next deadlines for the special issue.

Hope this is of help …

I'll aim for 24th Feb. Will that be OK?

- 2 March 2012

  Dear Olaf,
  … I'm attaching my manuscript for this article. As you will see, it's now the 2nd of March; (perhaps in error) I took a couple of extra days to incorporate feedback from some reviewers.

- 12 June 2012

  Dear Andrew
  Please find enclosed 3 review reports. One is referring to a marked copy of the pdf file, which is given as a separate attachment.
  We would ask you to give us the final version of your paper by August first.…we would like you to revise the paper, taking the comments you got into consideration, and give us a very brief summary of the main points revised.

- 3 August 2012

  Dear Olaf,
  It's 3rd August, and I wanted to give you an update on the status of this paper.
  I have responded to all of the reviewers' comments, and have annotated the referees' reports explaining how I have dealt with each. I finished this work last weekend, and have spent this week proof-reading and correcting small typographical errors and mis-phrasings. I have also had a colleague give the manuscript a careful proof-reading, and he has just told me that he now has a marked-up manuscript that he will get to me at the start of next week.
  So, I now plan to implement his corrections, and get you clean a version that responds to all of the referees' comments by the middle of next week. I hope that this will meet your schedule…

  Dear Andrew,
  Thank you, that sounds good!

- 10 August 2012
  Dear Olaf,
  My colleague found lots of little inconsistencies, which I believe that I have fixed to the best of my ability. I'm enclosing a new copy of the Manuscript, and my responses to the detailed suggestions made by referees 1 and 2. Referee 3's markup of the manuscript was also very useful, but I just implemented most of those suggestions and didn't comment on them.
  Note that I have split-off the final part of Section 4 into a new Section, numbered 5. This means that the old sections 5–12 are now numbered 6–13. I have also added a new subsection 11.2, which falls between what was 10.1 and 10.2. I'm mentioning this to make it easier to compare the two versions.

- 14 August 2012

  Dear Andrew,

  Thanks again for the revised version. We now just have some minor final comments, and leave the final adjustments to you. …

- 15 August 2012

  Dear Olaf,

  Herewith a revised version …

11 October 2012

Dear Olaf,

What's the status of this article, indeed of the whole special issue?

Hi

We are collecting the required reports, and hopefully you will hear from us soon

- 6 February 2013

  Hi Andrew,

  Everything should be ready now and about to sent to the publishers. Co-editor Jan Arne Telle has been waiting for one final review and this has caused some delay …

- Accepted! 16 February 2013

  Dear Andrew,

  We are happy to inform you that your paper has been accepted by I&C. The journal will contact you shortly (probably over the course of the next week) in order to collect their final source files.

  On behalf of the special issue editors

  Olaf

- Notification by Journal: 1 March 2013

  Dear author,

  We are pleased to confirm acceptance of your paper for publication in the FCT 2011 Special Issue of Information and Computation, following the recommendation of special issue guest editors Jan Arne Telle, Olaf Owe and Martin Steffen.

  Instructions for preparation of final source materials can be found here:

  http://www.elsevier.com/wps/find/journaldescription.cws_home/622844/authorinstructions

  The publisher's LaTeX style files can be found at the link below. The publisher prefers that you use the document class elsarticle.cls.

  http://www.elsevier.com/wps/find/authorsview.authors/elsarticle

  Please send the final source files and a printable copy of your paper to our office at iandc@csail.mit.edu.

- 29 May 2013

  To: Editorial Assistant, Information & Computation, MIT

  I tried to track this paper in the Elsevier online system. … But for that I apparently need a production reference number. What is it, please?

- 15 June 2013

  Hi Andrew,

  I'm still waiting on one remaining paper, and then all the files will be sent to the publisher at once. At that point, I believe they'll send you the reference number. Hopefully I'll be able to get everything to them within the next week or so, and the issue should appear online and in print shortly thereafter.

Summary Timeline

- Invited to Submit paper: 10 Nov 2011
- MS submitted: 2 March 2012
- Revisions submitted: 15 August 2012
- Accepted by *Information & Computation*
  - notification by Journal: 1 March 2013
- Deposit to ArXiv: 2 March 2013
- Transmission of MS to EIC: 2 March 2013

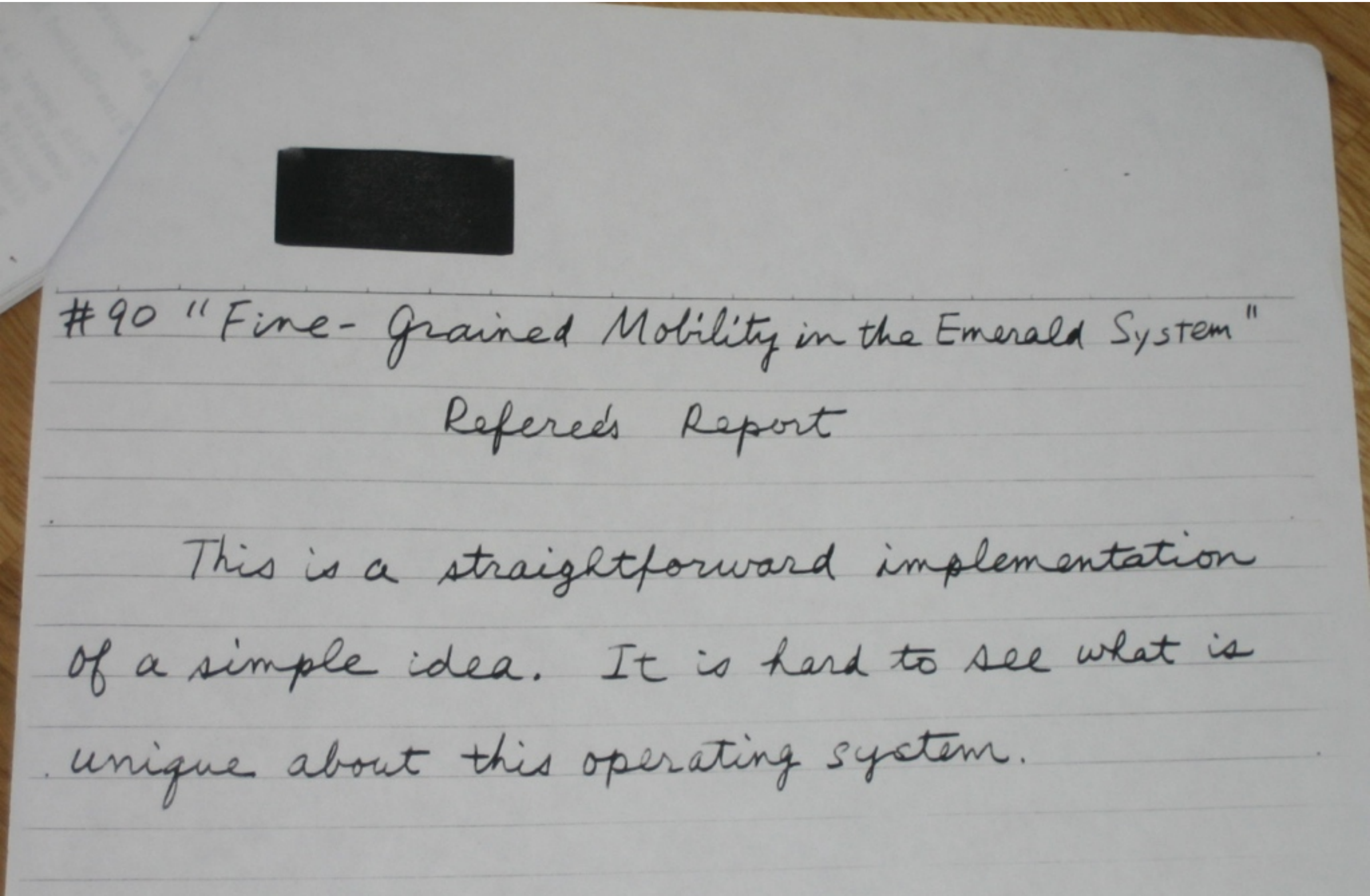
and then …

- 29 May 2013: asked for tracking number
- 2 August 2013: received "tracking number" from Elsevier
- 5 August: the proofs will be sent soon
- 5 August: choose Open Access or Subscription
    - 6 Aug: Andrew: I am willing to transfer copyright if the send me a suitable form
    - 7 Aug: Elsevier: my question had been forwarded to someone else
- 8 August: correct the proofs online within 48 hours
- 9 August: unformatted and unedited version available online
- 19 Aug: (automated) you haven't completed the publishing agreement …
- 26 October: last chance to order offprints (I didn't)
- 6 November: Congratulations on being published

Conference Publishing

Case Study

Conference Review, SOSP 1983



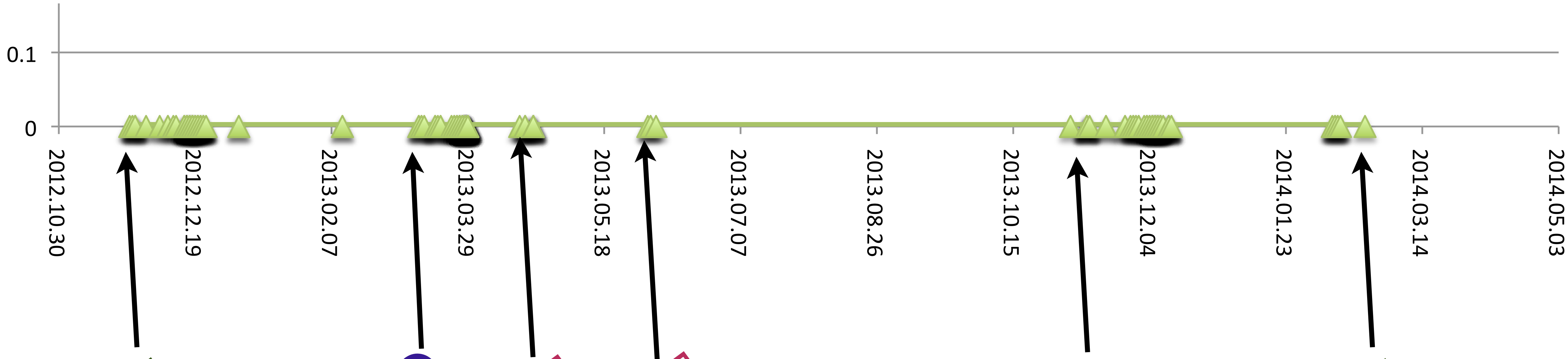
[redacted]

#90 "Fine-Grained Mobility in the Emerald System"

Referee's Report

This is a straightforward implementation of a simple idea. It is hard to see what is unique about this operating system.

Modules & Dialects in Grace



ECOOP 2013

OOPSLA 2013 (Dialects only)

DYLA 2013 (Modules only)

DYLA 2013 accepted; incorporate reviewer feedback

ECOOP 2014

ECOOP 2014 accepted

295 commits	over 15 months
black	66
mwh	156
kjx	58
zimothy	4
kim	11

Modules & Dialects in Grace

r768: 25 Nov 2012:

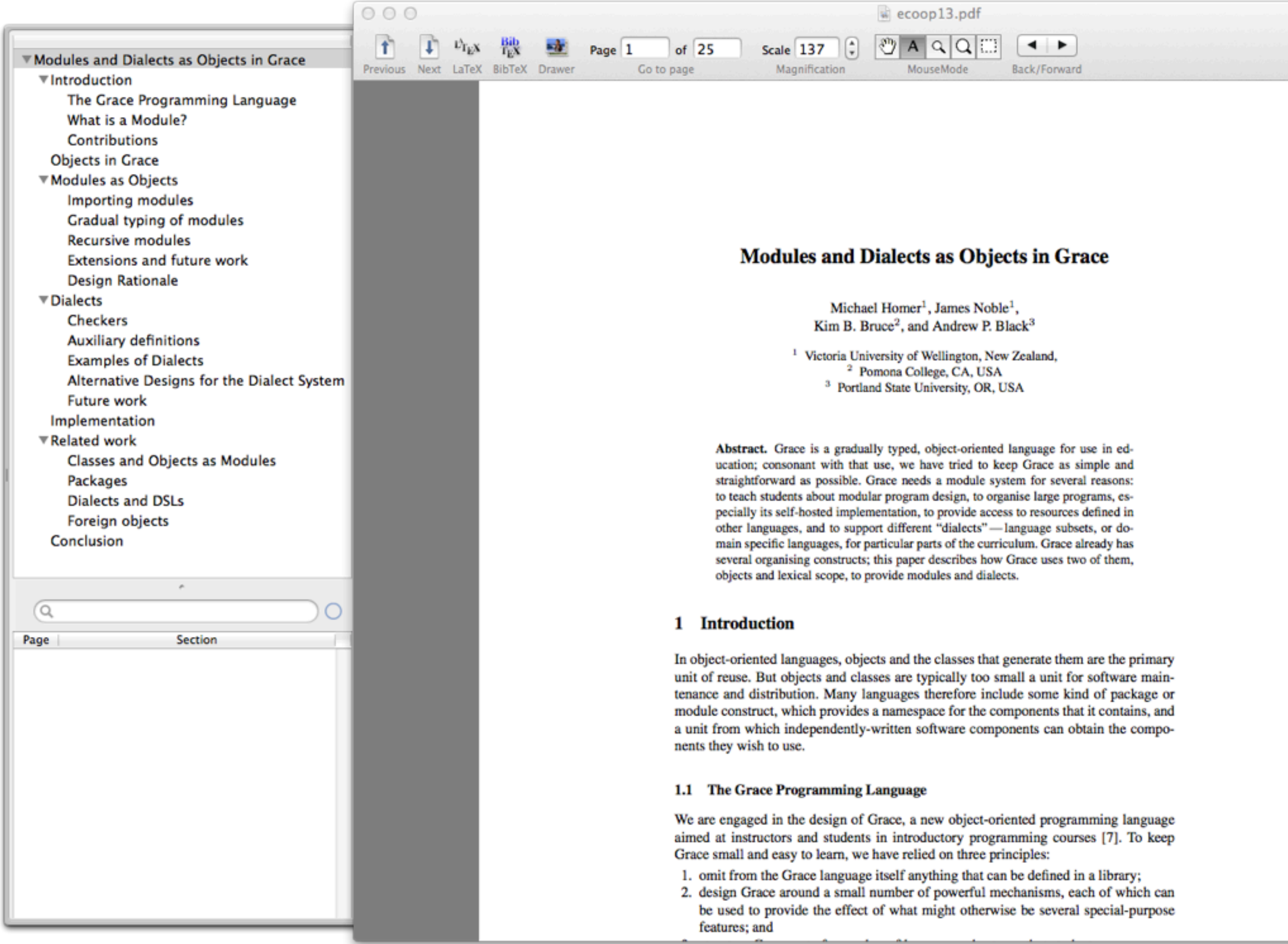
mwh created template for ECOOP 2013 Modules & Dialects paper

r779-868 26 Nov to 18 Dec:

mwh, kjx and black fill-out content

r902: 22 Dec 2012:

Submitted to ECOOP 2013



Modules and Dialects as Objects in Grace

Michael Homer[1], James Noble[1],
Kim B. Bruce[2], and Andrew P. Black[3]

[1] Victoria University of Wellington, New Zealand,
[2] Pomona College, CA, USA
[3] Portland State University, OR, USA

Abstract. Grace is a gradually typed, object-oriented language for use in education; consonant with that use, we have tried to keep Grace as simple and straightforward as possible. Grace needs a module system for several reasons: to teach students about modular program design, to organise large programs, especially its self-hosted implementation, to provide access to resources defined in other languages, and to support different "dialects"— language subsets, or domain specific languages, for particular parts of the curriculum. Grace already has several organising constructs; this paper describes how Grace uses two of them, objects and lexical scope, to provide modules and dialects.

1 Introduction

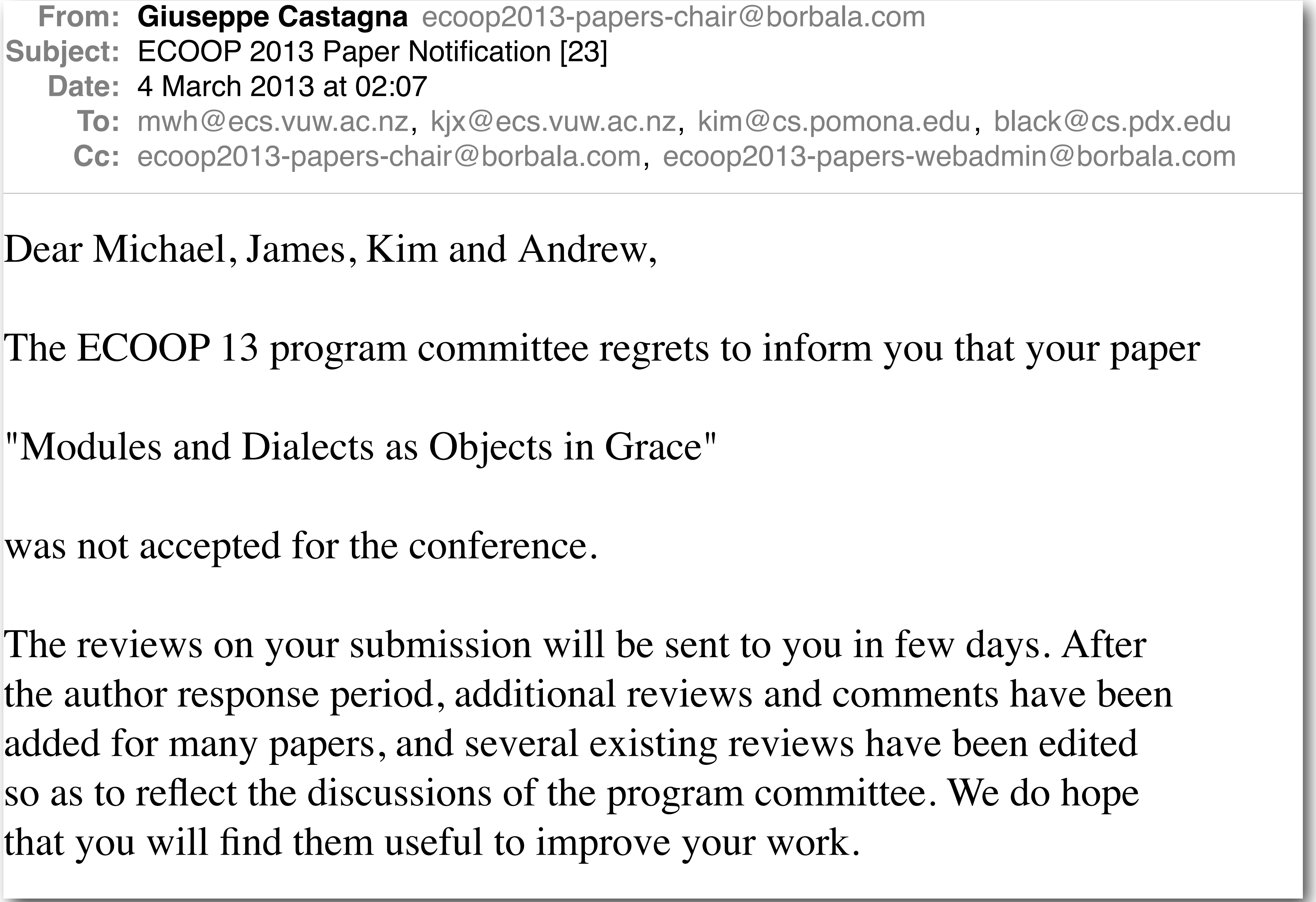
In object-oriented languages, objects and the classes that generate them are the primary unit of reuse. But objects and classes are typically too small a unit for software maintenance and distribution. Many languages therefore include some kind of package or module construct, which provides a namespace for the components that it contains, and a unit from which independently-written software components can obtain the components they wish to use.

1.1 The Grace Programming Language

We are engaged in the design of Grace, a new object-oriented programming language aimed at instructors and students in introductory programming courses [7]. To keep Grace small and easy to learn, we have relied on three principles:

1. omit from the Grace language itself anything that can be defined in a library;
2. design Grace around a small number of powerful mechanisms, each of which can be used to provide the effect of what might otherwise be several special-purpose features; and

Rejected



From: **Giuseppe Castagna** ecoop2013-papers-chair@borbala.com
Subject: ECOOP 2013 Paper Notification [23]
Date: 4 March 2013 at 02:07
To: mwh@ecs.vuw.ac.nz, kjx@ecs.vuw.ac.nz, kim@cs.pomona.edu, black@cs.pdx.edu
Cc: ecoop2013-papers-chair@borbala.com, ecoop2013-papers-webadmin@borbala.com

Dear Michael, James, Kim and Andrew,

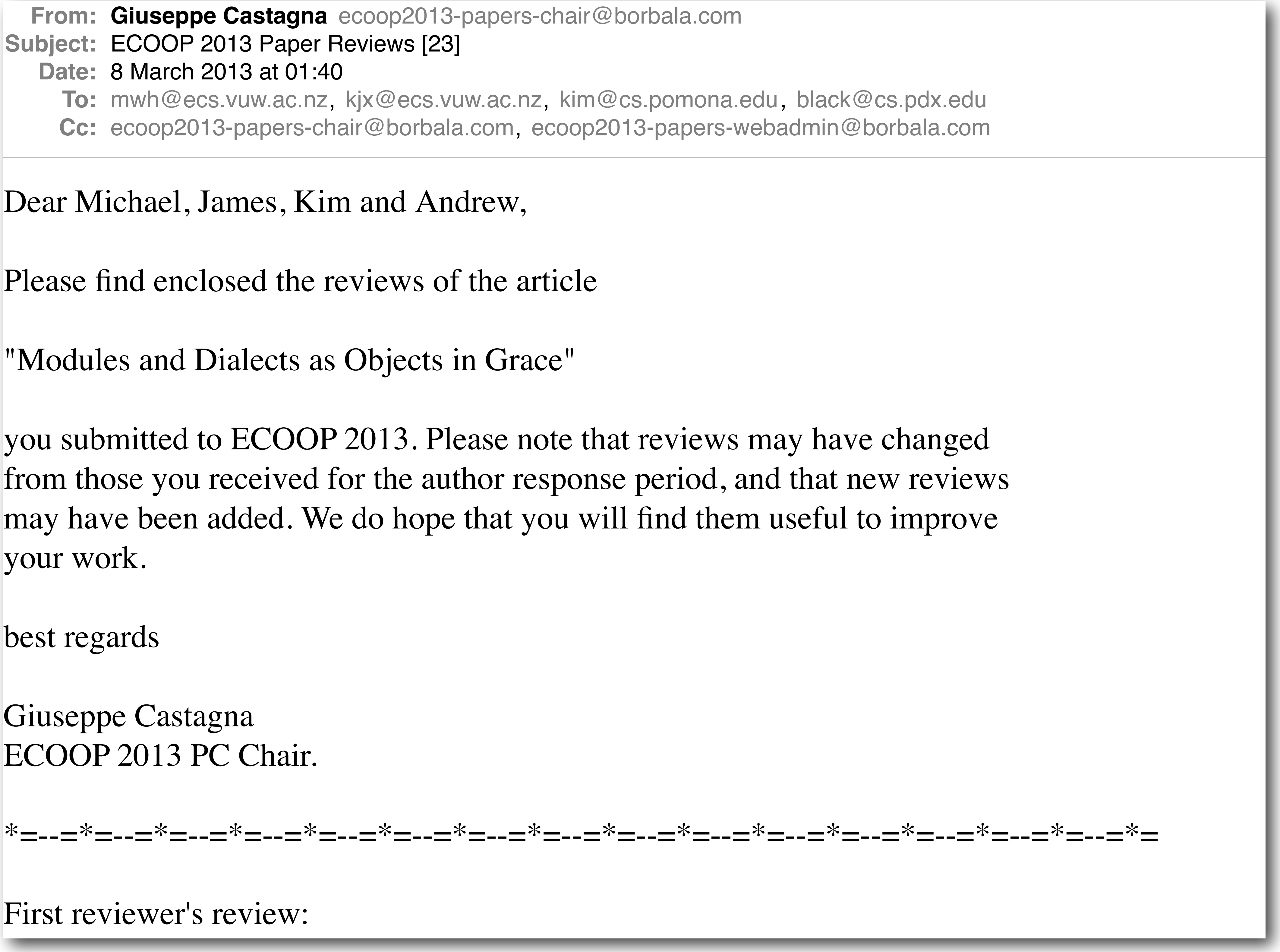
The ECOOP 13 program committee regrets to inform you that your paper

"Modules and Dialects as Objects in Grace"

was not accepted for the conference.

The reviews on your submission will be sent to you in few days. After the author response period, additional reviews and comments have been added for many papers, and several existing reviews have been edited so as to reflect the discussions of the program committee. We do hope that you will find them useful to improve your work.

Reviewers' Comments



From: **Giuseppe Castagna** ecoop2013-papers-chair@borbala.com
Subject: ECOOP 2013 Paper Reviews [23]
Date: 8 March 2013 at 01:40
To: mwh@ecs.vuw.ac.nz, kjx@ecs.vuw.ac.nz, kim@cs.pomona.edu, black@cs.pdx.edu
Cc: ecoop2013-papers-chair@borbala.com, ecoop2013-papers-webadmin@borbala.com

Dear Michael, James, Kim and Andrew,

Please find enclosed the reviews of the article

"Modules and Dialects as Objects in Grace"

you submitted to ECOOP 2013. Please note that reviews may have changed from those you received for the author response period, and that new reviews may have been added. We do hope that you will find them useful to improve your work.

best regards

Giuseppe Castagna
ECOOP 2013 PC Chair.

=--==--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=--=*=

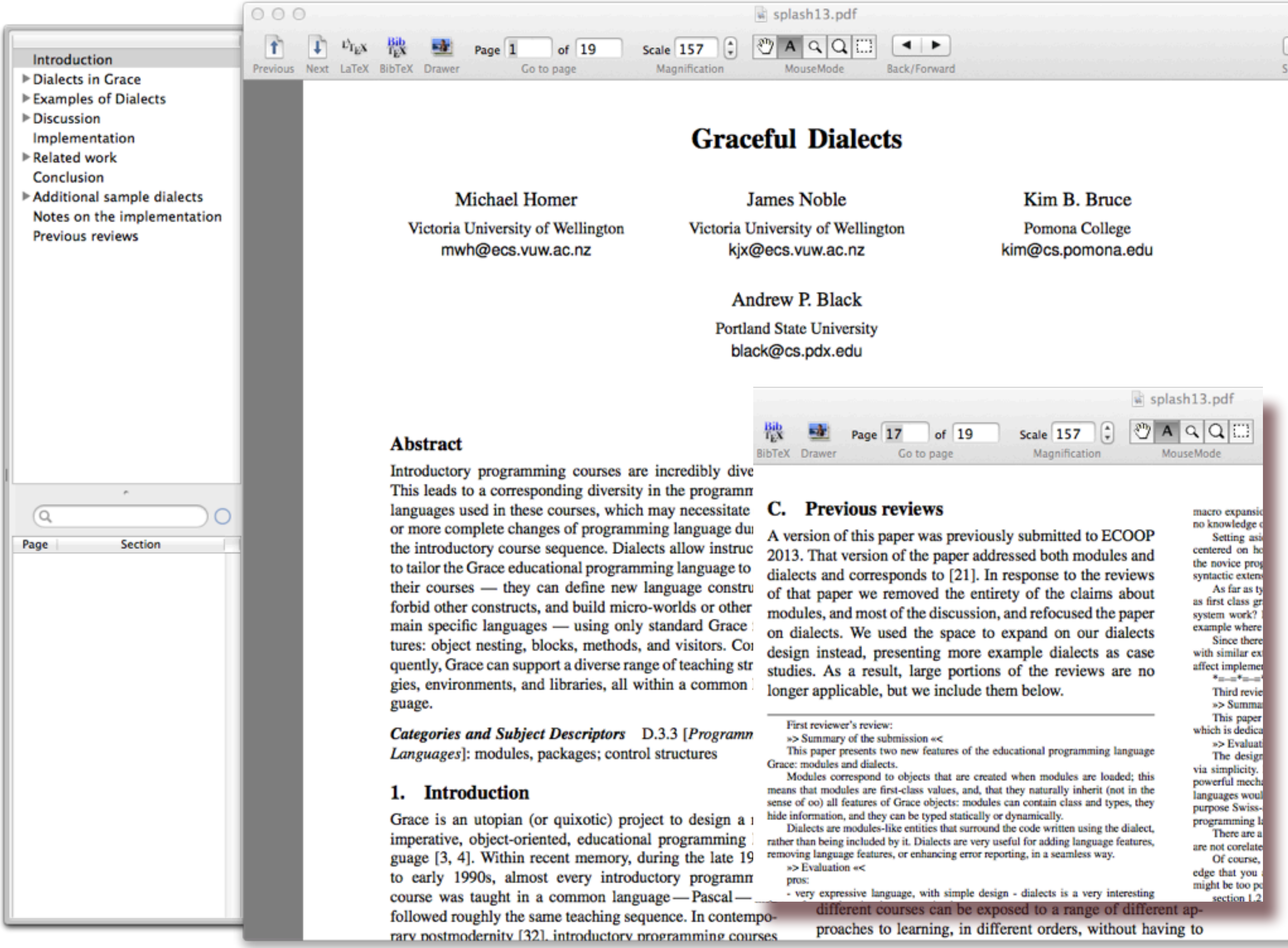
First reviewer's review:

Six reviews, 4300 words

- Several reviewers had issues understanding what we meant
  - “I found the motivations, though clearly enumerated, a bit confusing.”
  - “It remains unclear what the novel contributions are, maybe its single language feature, or a new combination that allows for new interesting characteristics, or resolves some trade-off... Whatever it is, it remains unclear. The reader can only speculate about it.”
  - “the paper and its organization make it hard to easily understand novel aspect and interesting ideas. The paper forces readers to read the complete story from beginning to end in order to figure out what might be new or interesting about Grace and its module system (which remains unclear)”
  - “I had difficulties to fully understand your idea of dialects, still unsure whether I got it correctly. Maybe illustrations of your mental model could help, if you have something like this.”
  - “When reading it, I don’t understand R4 at all.”
  - “what do you mean with "a future program development system could represent modules differently"? Do you mean a subsystem (or submodule)?”
  - “the paper does not allow the reader to clearly identify novel aspects.”
  - “the paper lacks clear, precise descriptions and examples to support your argument. For example, is it really true that Newspeak modules require much redundant boilerplate code? Why’s that? If there really is a problem, you should explain it in detail (and maybe provide code examples, some analysis of written Newspeak code).”

Revise & Resubmit

- 98 revisions later …
  - re-focus on dialects only
- 2013 March 29:
  - Submitted to OOPSLA



Graceful Dialects

Michael Homer
Victoria University of Wellington
mwh@ecs.vuw.ac.nz

James Noble
Victoria University of Wellington
kjx@ecs.vuw.ac.nz

Kim B. Bruce
Pomona College
kim@cs.pomona.edu

Andrew P. Black
Portland State University
black@cs.pdx.edu

Abstract

Introductory programming courses are incredibly dive
This leads to a corresponding diversity in the programn
languages used in these courses, which may necessitate
or more complete changes of programming language du
the introductory course sequence. Dialects allow instruc
to tailor the Grace educational programming language to
their courses — they can define new language constru
forbid other constructs, and build micro-worlds or other
main specific languages — using only standard Grace
tures: object nesting, blocks, methods, and visitors. Co
quently, Grace can support a diverse range of teaching str
gies, environments, and libraries, all within a common
guage.

Categories and Subject Descriptors D.3.3 [*Programm
Languages*]: modules, packages; control structures

1. Introduction

Grace is an utopian (or quixotic) project to design a
imperative, object-oriented, educational programming
guage [3, 4]. Within recent memory, during the late 19
to early 1990s, almost every introductory programm
course was taught in a common language — Pascal —
followed roughly the same teaching sequence. In contempo-
rary postmodernity [32], introductory programming courses

different courses can be exposed to a range of different ap-
proaches to learning, in different orders, without having to

C. Previous reviews

A version of this paper was previously submitted to ECOOP 2013. That version of the paper addressed both modules and dialects and corresponds to [21]. In response to the reviews of that paper we removed the entirety of the claims about modules, and most of the discussion, and refocused the paper on dialects. We used the space to expand on our dialects design instead, presenting more example dialects as case studies. As a result, large portions of the reviews are no longer applicable, but we include them below.

First reviewer's review:

»> Summary of the submission «<

This paper presents two new features of the educational programming language Grace: modules and dialects.

Modules correspond to objects that are created when modules are loaded; this means that modules are first-class values, and, that they naturally inherit (not in the sense of oo) all features of Grace objects: modules can contain class and types, they hide information, and they can be typed statically or dynamically.

Dialects are modules-like entities that surround the code written using the dialect, rather than being included by it. Dialects are very useful for adding language features, removing language features, or enhancing error reporting, in a seamless way.

»> Evaluation «<

pros:

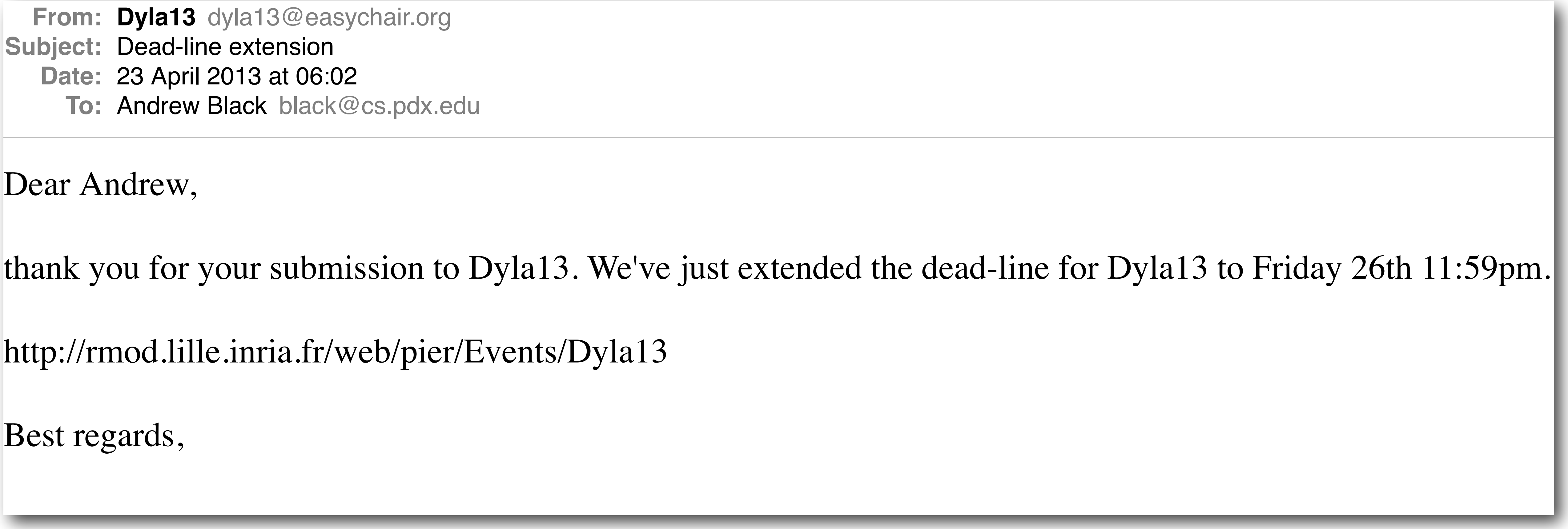
- very expressive language, with simple design - dialects is a very interesting

macro expansio
no knowledge o
Setting asi
centered on ho
the novice prog
syntactic extens
As far as ty
as first class gr
system work?
example where
Since there
with similar ex
affect implemen
=-==-=*
Third revie
»> Summa
This paper
which is dedica
»> Evaluati
The design
via simplicity.
powerful mecha
languages woul
purpose Swiss-
programming la
There are a
are not corelate
Of course,
edge that you
might be too po
section 1.2

Modules as Objects

2013.04.17 — 2013.04.22

- extract modules material into new paper
- submit to Dyla’13 (Dynamic Languages and Applications), a satellite meeting at ECOOP



From: **Dyla13** dyla13@easychair.org
Subject: Dead-line extension
Date: 23 April 2013 at 06:02
To: Andrew Black black@cs.pdx.edu

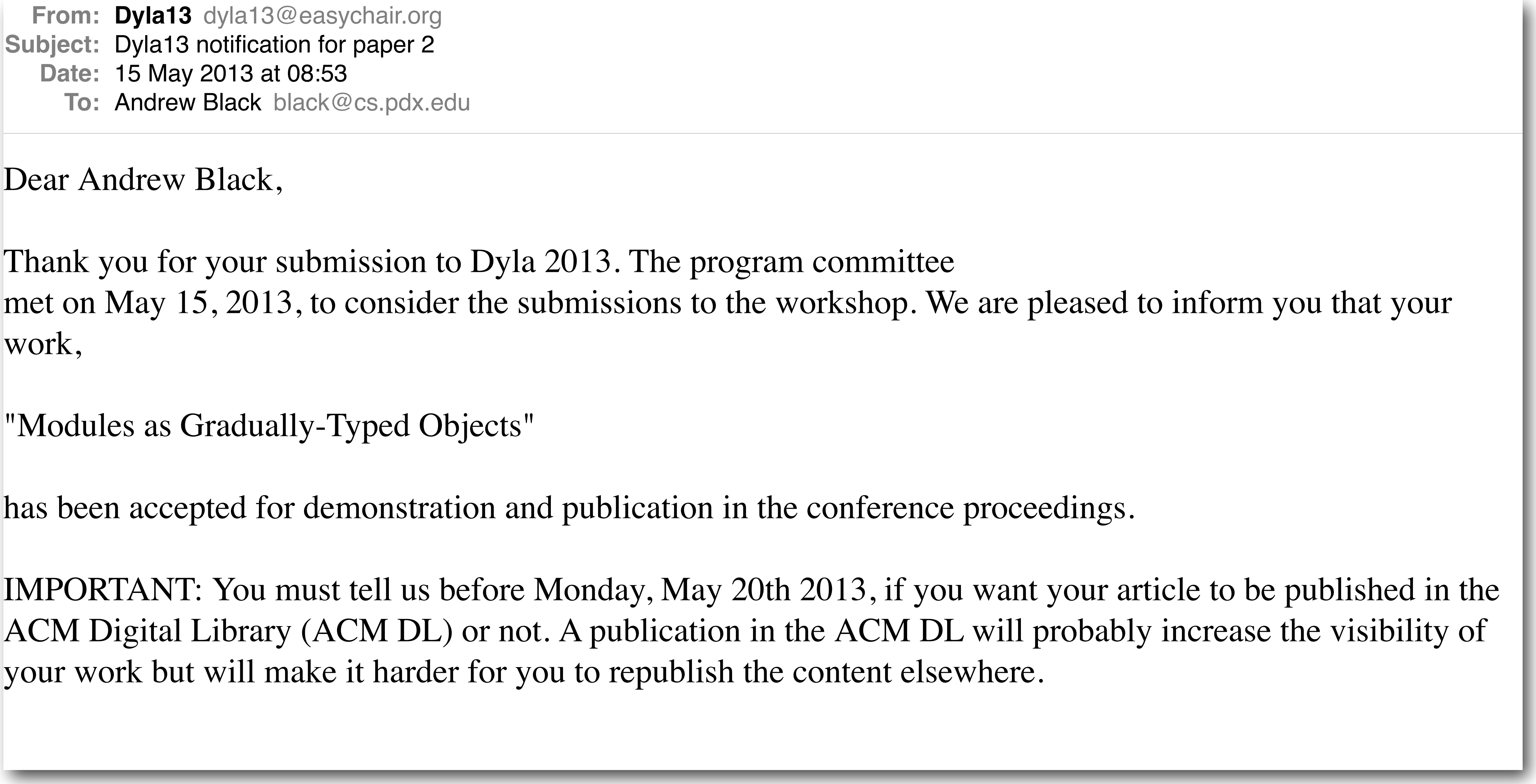
Dear Andrew,

thank you for your submission to Dyla13. We've just extended the dead-line for Dyla13 to Friday 26th 11:59pm.

http://rmod.lille.inria.fr/web/pier/Events/Dyla13

Best regards,

Dyla’13 Modules: Accepted!



From: **Dyla13** dyla13@easychair.org
Subject: Dyla13 notification for paper 2
Date: 15 May 2013 at 08:53
To: Andrew Black black@cs.pdx.edu

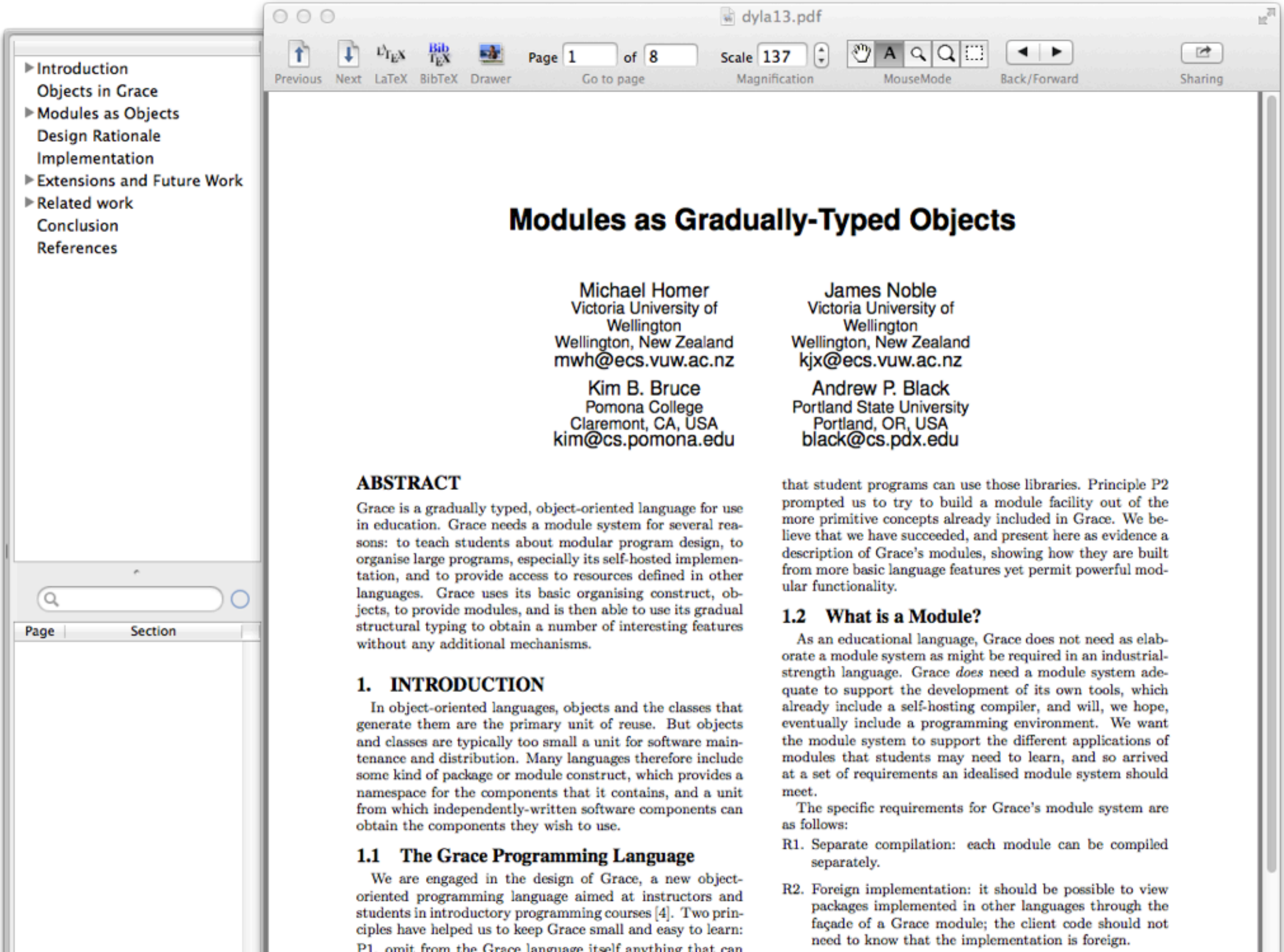
Dear Andrew Black,

Thank you for your submission to Dyla 2013. The program committee
met on May 15, 2013, to consider the submissions to the workshop. We are pleased to inform you that your work,

"Modules as Gradually-Typed Objects"

has been accepted for demonstration and publication in the conference proceedings.

IMPORTANT: You must tell us before Monday, May 20th 2013, if you want your article to be published in the ACM Digital Library (ACM DL) or not. A publication in the ACM DL will probably increase the visibility of your work but will make it harder for you to republish the content elsewhere.



Modules as Gradually-Typed Objects

Michael Homer
Victoria University of Wellington
Wellington, New Zealand
mwh@ecs.vuw.ac.nz

James Noble
Victoria University of Wellington
Wellington, New Zealand
kjx@ecs.vuw.ac.nz

Kim B. Bruce
Pomona College
Claremont, CA, USA
kim@cs.pomona.edu

Andrew P. Black
Portland State University
Portland, OR, USA
black@cs.pdx.edu

ABSTRACT

Grace is a gradually typed, object-oriented language for use in education. Grace needs a module system for several reasons: to teach students about modular program design, to organise large programs, especially its self-hosted implementation, and to provide access to resources defined in other languages. Grace uses its basic organising construct, objects, to provide modules, and is then able to use its gradual structural typing to obtain a number of interesting features without any additional mechanisms.

1. INTRODUCTION

In object-oriented languages, objects and the classes that generate them are the primary unit of reuse. But objects and classes are typically too small a unit for software maintenance and distribution. Many languages therefore include some kind of package or module construct, which provides a namespace for the components that it contains, and a unit from which independently-written software components can obtain the components they wish to use.

1.1 The Grace Programming Language

We are engaged in the design of Grace, a new object-oriented programming language aimed at instructors and students in introductory programming courses [4]. Two principles have helped us to keep Grace small and easy to learn: P1. omit from the Grace language itself anything that can

that student programs can use those libraries. Principle P2 prompted us to try to build a module facility out of the more primitive concepts already included in Grace. We believe that we have succeeded, and present here as evidence a description of Grace's modules, showing how they are built from more basic language features yet permit powerful modular functionality.

1.2 What is a Module?

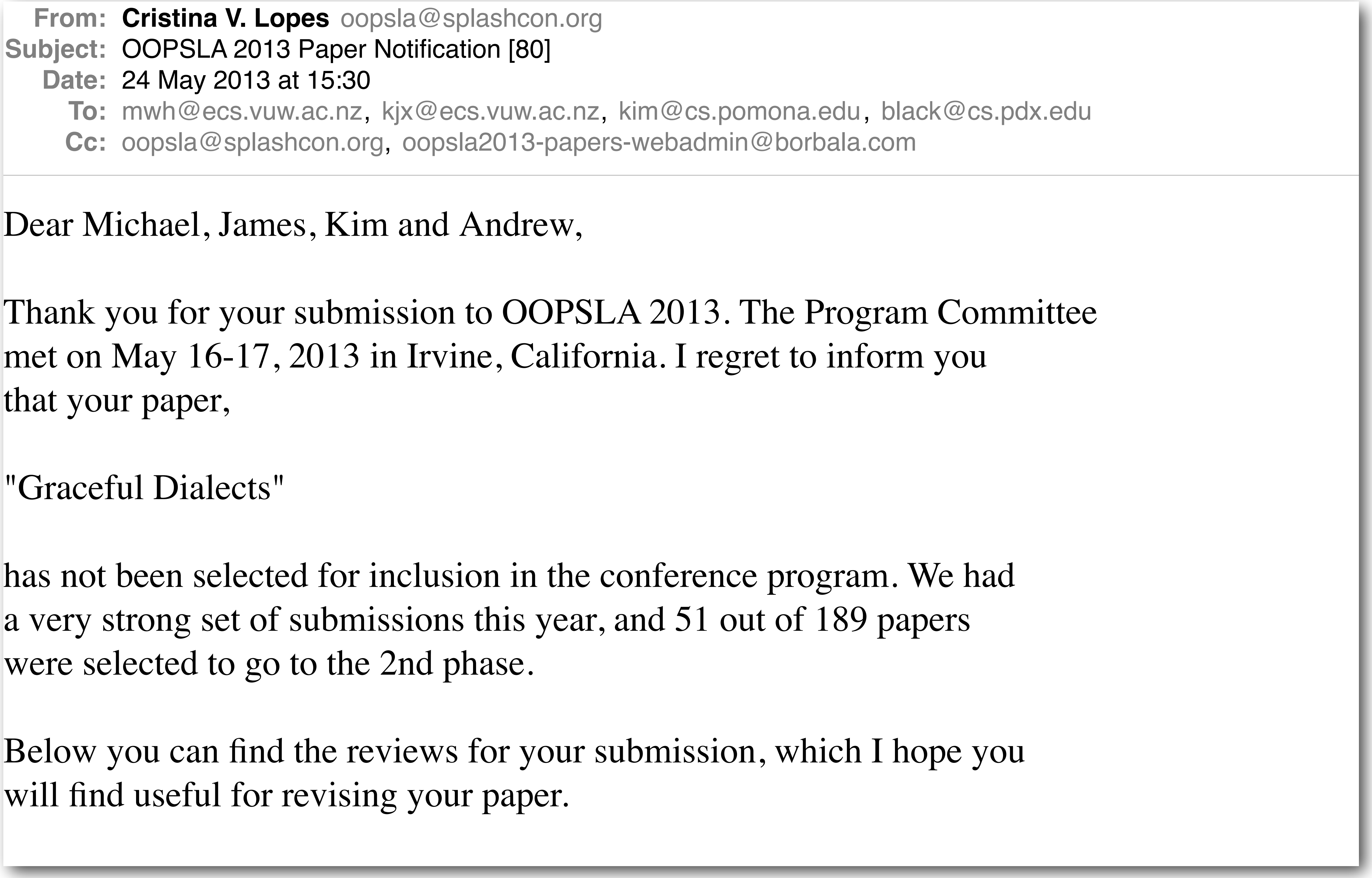
As an educational language, Grace does not need as elaborate a module system as might be required in an industrial-strength language. Grace *does* need a module system adequate to support the development of its own tools, which already include a self-hosting compiler, and will, we hope, eventually include a programming environment. We want the module system to support the different applications of modules that students may need to learn, and so arrived at a set of requirements an idealised module system should meet.

The specific requirements for Grace's module system are as follows:

R1. Separate compilation: each module can be compiled separately.

R2. Foreign implementation: it should be possible to view packages implemented in other languages through the façade of a Grace module; the client code should not need to know that the implementation is foreign.

OOPSLA Dialects: Rejected



From: **Cristina V. Lopes** oopsla@splashcon.org
Subject: OOPSLA 2013 Paper Notification [80]
Date: 24 May 2013 at 15:30
To: mwh@ecs.vuw.ac.nz, kjx@ecs.vuw.ac.nz, kim@cs.pomona.edu, black@cs.pdx.edu
Cc: oopsla@splashcon.org, oopsla2013-papers-webadmin@borbala.com

Dear Michael, James, Kim and Andrew,

Thank you for your submission to OOPSLA 2013. The Program Committee
met on May 16-17, 2013 in Irvine, California. I regret to inform you
that your paper,

"Graceful Dialects"

has not been selected for inclusion in the conference program. We had
a very strong set of submissions this year, and 51 out of 189 papers
were selected to go to the 2nd phase.

Below you can find the reviews for your submission, which I hope you
will find useful for revising your paper.

Revise for ECOOP 2014

r1326–r1346:   5–25 Nov 2013

- Examined reviews carefully
- Edited them into the source as `\ri{…}` comments
- Wrote new introduction.

…

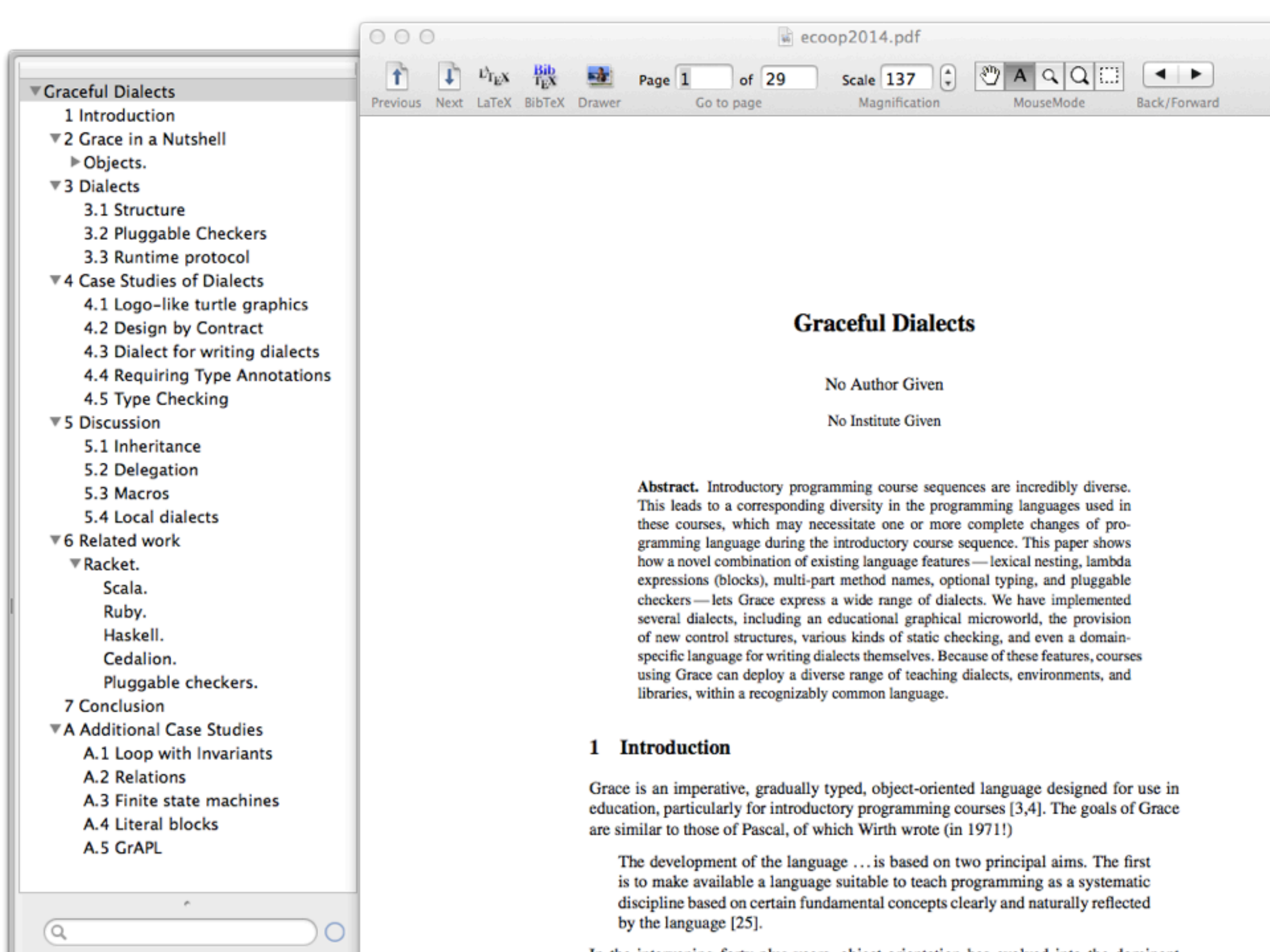
r1416:   8 Dec 2013

- Exchanged sections 4.3 and 4.4

…

r1433: 12 Dec 2013

- Submitted!



Graceful Dialects

No Author Given

No Institute Given

Abstract. Introductory programming course sequences are incredibly diverse. This leads to a corresponding diversity in the programming languages used in these courses, which may necessitate one or more complete changes of programming language during the introductory course sequence. This paper shows how a novel combination of existing language features — lexical nesting, lambda expressions (blocks), multi-part method names, optional typing, and pluggable checkers — lets Grace express a wide range of dialects. We have implemented several dialects, including an educational graphical microworld, the provision of new control structures, various kinds of static checking, and even a domain-specific language for writing dialects themselves. Because of these features, courses using Grace can deploy a diverse range of teaching dialects, environments, and libraries, within a recognizably common language.

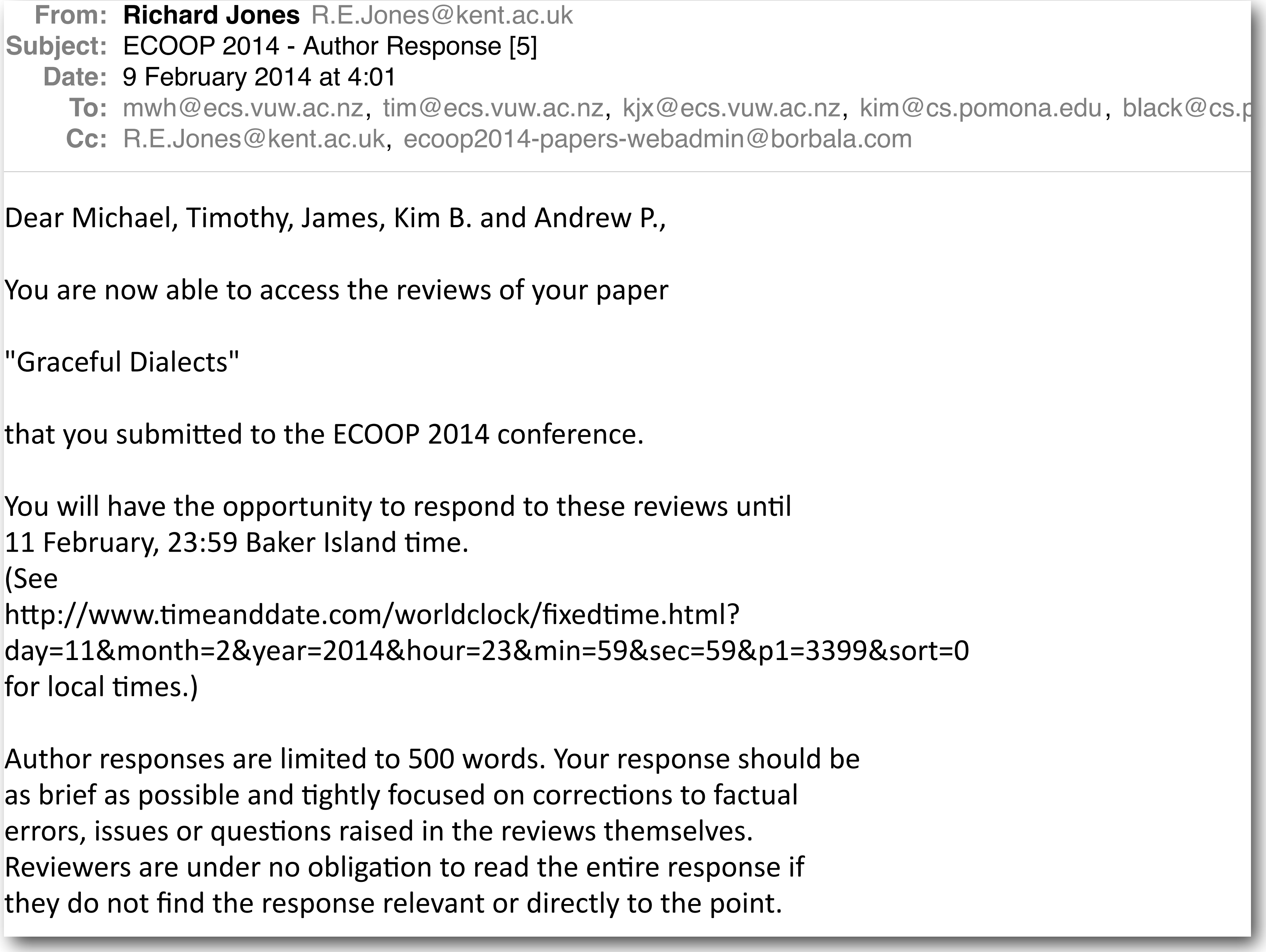
1 Introduction

Grace is an imperative, gradually typed, object-oriented language designed for use in education, particularly for introductory programming courses [3,4]. The goals of Grace are similar to those of Pascal, of which Wirth wrote (in 1971!)

> The development of the language . . . is based on two principal aims. The first is to make available a language suitable to teach programming as a systematic discipline based on certain fundamental concepts clearly and naturally reflected by the language [25].

Author Response

9 Feb 2013: Author Response invited



From: **Richard Jones** R.E.Jones@kent.ac.uk
Subject: ECOOP 2014 - Author Response [5]
Date: 9 February 2014 at 4:01
To: mwh@ecs.vuw.ac.nz, tim@ecs.vuw.ac.nz, kjx@ecs.vuw.ac.nz, kim@cs.pomona.edu, black@cs.p
Cc: R.E.Jones@kent.ac.uk, ecoop2014-papers-webadmin@borbala.com

Dear Michael, Timothy, James, Kim B. and Andrew P.,

You are now able to access the reviews of your paper

"Graceful Dialects"

that you submitted to the ECOOP 2014 conference.

You will have the opportunity to respond to these reviews until
11 February, 23:59 Baker Island time.
(See
http://www.timeanddate.com/worldclock/fixedtime.html?day=11&month=2&year=2014&hour=23&min=59&sec=59&p1=3399&sort=0
for local times.)

Author responses are limited to 500 words. Your response should be as brief as possible and tightly focused on corrections to factual errors, issues or questions raised in the reviews themselves. Reviewers are under no obligation to read the entire response if they do not find the response relevant or directly to the point.

Reviews identified several issues …

Our Response:

We thank the reviewers for their comments and suggestions, which we appreciate. We confine our response to two areas: applicability, and evaluation.

====Applicability====

The reviews raise the question: does our approach to dialects have general application beyond the Grace language?

The novel contribution of the paper is a system to extend and restrict the language available to a particular program or part thereof, without introducing new semantics or concepts. This is generally applicable, not just to teaching languages, but to any situation where a language is extended via a library, or where there is a need to restrict use of built-in features. Our approach is transferable to, and useful in, other languages: nothing in the problem, nor the solution, is specific to Grace.

The ability to integrate domain-specific languages into existing languages, and together into a single program with uniform semantics, without relying on a complex and specialised type system, is generally useful. In addition to DSLs, systems such as design-by-contract that require programs to be organised in new ways can be implemented using our design. Checkers can implement a range of novel type systems, broadly construed, wholly within an existing language.

We will modify the introduction and language summary, particularly the contribution paragraph identified by the first reviewer, to clarify these points.

Portland State
UNIVERSITY

Author Response (continued):

====Evaluation====

We believe that Grace dialects should be evaluated as contribution to language design, and not by their success in teaching programming to novices. Indeed, our approach to implementing dialects can succeed or fail independently of the success or failure of Grace as a teaching language.

Evidence for the success of our approach to dialects comes in two forms: (1) an implemented (public, self-hosted) system, and (2) a series of case studies. We note this passage from the call for papers:

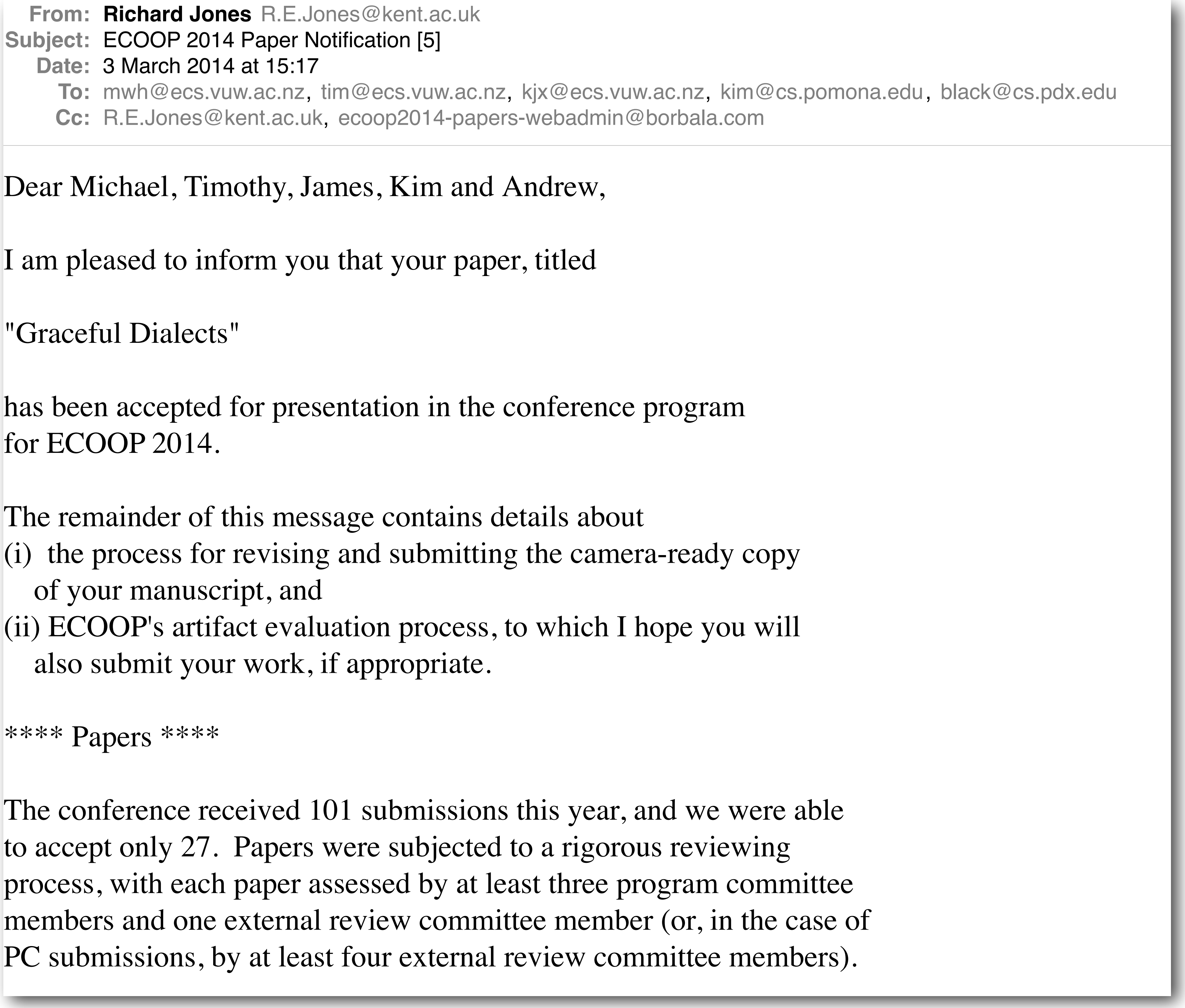
> "The paper presents evidence supporting its claims. Examples of evidence include formalizations and proofs, implemented systems, experimental results, statistical analyses and case studies."

We agree that classroom evaluation of Grace will be useful. Indeed, it is an ongoing project that will likely be the subject of future ECOOP and SIGCSE papers. However, classroom use is not an appropriate way to evaluate a general-purpose technical contribution. Instead, we have evaluated the technical contribution of our paper — a language design and implementation technique that applies to a wide range of languages, including those not intended for teaching — using evaluation mechanisms we feel are appropriate for that purpose.

If ECOOP were to decide that every language design paper must include the results of an extended evaluation of the novel features in an industrial setting, then design papers would cease to appear at ECOOP. Such evaluations are enormously useful, but they are not a pre-requisite to publishing a description and a technical evaluation of a novel design and implementation.

These reviews have made us realise that the paper needs to more clearly distinguish the evaluation of dialects from the broader goals of the Grace project to improve teaching. We will clarify the wording of the introduction accordingly.

3 March 2014: Accepted!



From: **Richard Jones** R.E.Jones@kent.ac.uk
Subject: ECOOP 2014 Paper Notification [5]
Date: 3 March 2014 at 15:17
To: mwh@ecs.vuw.ac.nz, tim@ecs.vuw.ac.nz, kjx@ecs.vuw.ac.nz, kim@cs.pomona.edu, black@cs.pdx.edu
Cc: R.E.Jones@kent.ac.uk, ecoop2014-papers-webadmin@borbala.com

Dear Michael, Timothy, James, Kim and Andrew,

I am pleased to inform you that your paper, titled

"Graceful Dialects"

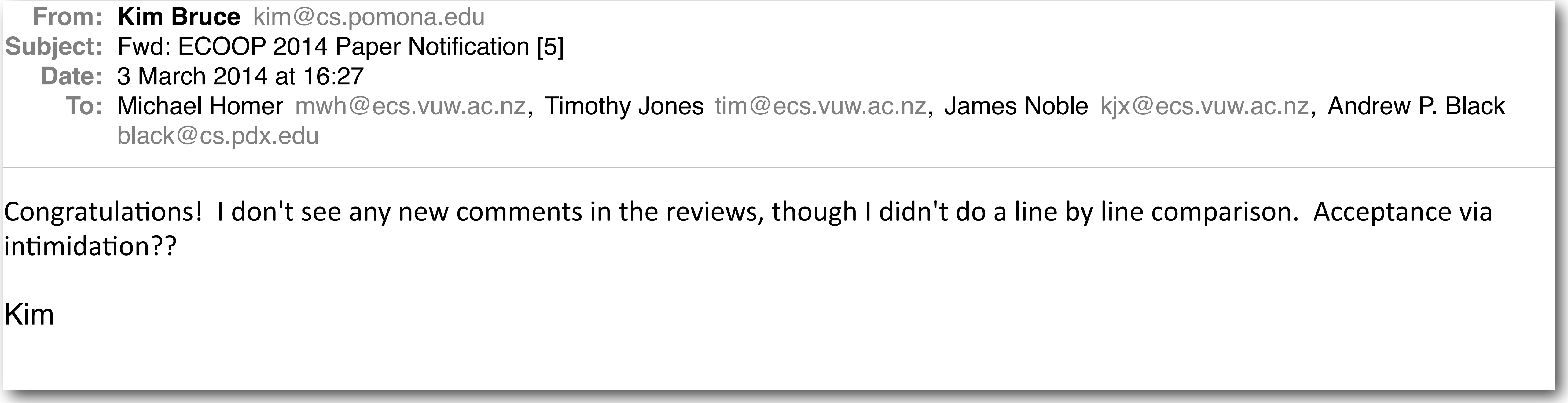
has been accepted for presentation in the conference program
for ECOOP 2014.

The remainder of this message contains details about
(i) the process for revising and submitting the camera-ready copy
of your manuscript, and
(ii) ECOOP's artifact evaluation process, to which I hope you will
also submit your work, if appropriate.

**** Papers ****

The conference received 101 submissions this year, and we were able to accept only 27. Papers were subjected to a rigorous reviewing process, with each paper assessed by at least three program committee members and one external review committee member (or, in the case of PC submissions, by at least four external review committee members).

The last word

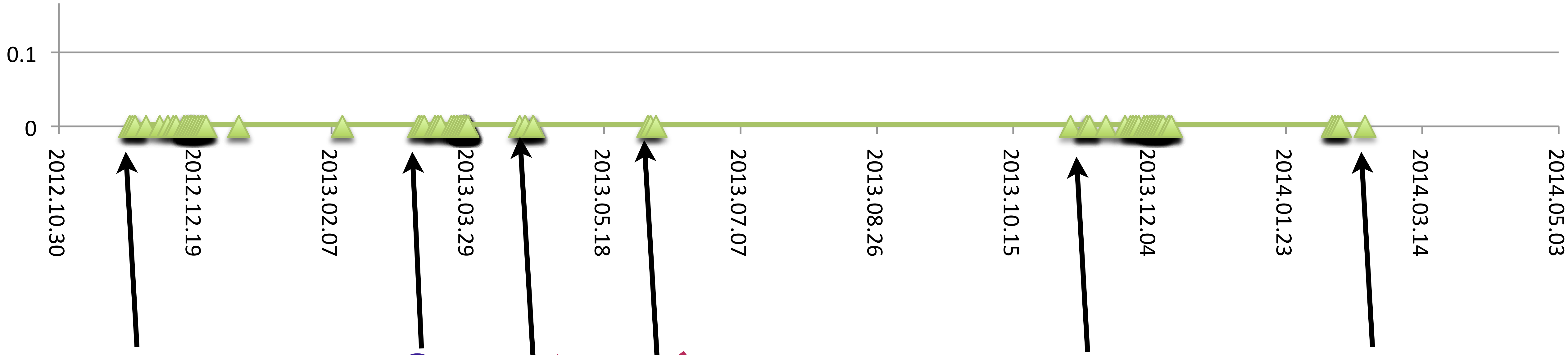


From: **Kim Bruce** kim@cs.pomona.edu
Subject: Fwd: ECOOP 2014 Paper Notification [5]
Date: 3 March 2014 at 16:27
To: Michael Homer mwh@ecs.vuw.ac.nz, Timothy Jones tim@ecs.vuw.ac.nz, James Noble kjx@ecs.vuw.ac.nz, Andrew P. Black black@cs.pdx.edu

Congratulations! I don't see any new comments in the reviews, though I didn't do a line by line comparison. Acceptance via intimidation??

Kim

Modules & Dialects in Grace



ECOOP 2013

OOPSLA 2013 (Dialects only)

DYLA 2013 (Modules only)

DYLA 2013 accepted; incorporate reviewer feedback

ECOOP 2014

ECOOP 2014 accepted

295 commits	over 15 months
black	66
mwh	156
kjx	58
zimothy	4
kim	11