

# Abstract

Progress in the field of wireless technology has resulted in a growing body of research that deals with *mobile ad hoc networks* (MANETs): networks composed of *mobile* devices that are connected by *wireless* communication links with a limited communication range. The limited communication range of devices combined with the fact that devices move about renders applications subject to higher rates of partial failures than in traditional, stationary networks. This dissertation investigates programming language support to deal with the effects engendered by *partial failures* in the software development process. In particular, we investigate support for handling partial failures in MANET applications, from the programming language design to tool support.

In order to support distributed programming in such a dynamically changing environment, it is worth to investigate software engineering techniques where network disconnections are not treated as the exception but the rule. This observation motivated the development of the *ambient-oriented programming paradigm* in which this work is rooted. Within the context of this paradigm, we propose expressive abstractions that aid developers to detect, reason about, and handle partial failures. Our survey of related work will reveal that *leasing* provides a solution for a failure handling model in MANETs. However, to date no leasing model has been designed specifically for MANETs. In this context, leasing needs to be reconciled with computational models that deal with transient failures, and provide a well-defined high-level interface to allow developers to handle failures in a modular and reusable way, while accommodating the various application needs. This work studies a failure handling model which combines leasing with ambient-oriented programming, leading to the concept of *ambient-oriented leasing*.

We investigate ambient-oriented leasing in the two distinct distributed computing models. First, we integrate our leasing model in a distributed object-oriented model as a novel referencing abstraction, called *leased object references*. Second, we explore the applicability of ambient-oriented leasing to tuple spaces, resulting in a novel adaptation of the tuple space model for MANETs called *TOTAM* (Tuples on the Ambient). We design and implement both incarnations of ambient-oriented leasing in *AmbientTalk/M*, an ambient-oriented language which provides a unified meta model on which both language abstractions as well as tool support is built.

To support the development of MANET applications, we study tool support in the form of an *ambient-oriented debugger* that helps programmers to achieve a better understanding of the dynamic behaviour of a MANET application. As a proof-of-concept tool we provide an online ambient-oriented debugger integrated into AmbientTalk Eclipse IDE called *REME-D* (Reflective Epidemic MESSage-oriented Debugger).

Together, the programming language abstractions and the debugging support presented in this dissertation provide an innovative toolbox that allows MANET application developers to deal with partial failures in an anticipated and controllable way.



# Samenvatting

Innovaties in het domein van draadloze technologie hebben geleid tot een onderzoeksstroming die zich toespitst op zogenaamde mobiele ad-hoc netwerken (MANETs): dit zijn netwerken waarbij *mobiele* apparaten met elkaar verbonden zijn door middel van draadloze communicatie-links die gelimiteerd zijn in hun reikwijdte. De gelimiteerde reikwijdte van de communicatie in combinatie met het feit dat deze toestellen mobiel, zijn leidt ertoe dat toepassingen die gebruik maken van zulke netwerken veel meer blootgesteld worden aan partiële fouten dan toepassingen die gebruik maken van een traditionele gedistribueerde netwerkinfrastructuur. Deze thesis onderzoekt programmeerabstracties die het mogelijk maken om om te springen met de nefaste effecten als gevolg van partiële fouten tijdens het software-ontwikkelingsproces. In het bijzonder onderzoeken we ondersteuning voor het omspringen met partiële fouten voor toepassingen die draaien op MANETs. Dit gaat van programmeertaalontwerp tot tool-ondersteuning.

Om ondersteuning te geven tijdens het programmeren van gedistribueerde applicaties in zulke dynamisch veranderende omgeving, is het nuttig om software engineering-technieken te onderzoeken waarbij het verbreken van een netwerkverbinding niet gezien wordt als een uitzondering maar eerder als de regel. Deze observatie heeft de ontwikkeling van het *ambient-georiënteerd paradigma* gemotiveerd waar ook dit werk zijn wortels vindt. In de context van dit paradigma stellen wij expressieve abstracties voor die de ontwikkelaar helpen om partiële fouten te detecteren, erover te redeneren, en ermee om te springen. Ons overzicht van gerelateerd werk zal onthullen dat *leasing* een oplossing biedt als basis voor een foutenaafhandelsmodel in MANETs. Echter, tot op heden is er geen leasing-model ontwikkeld specifiek voor MANETs. In deze context moet leasing verzoent worden met modellen die omspringen met tijdelijke fouten, en moet er een welgedefinieerde interface van een voldoende hoog abstratieniveau aangeboden worden. Dit moet de programmeur toelaten om te springen met fouten in een modulaire en herbruikbare manier, zonder de verschillende software-vereisten uit het oog te verliezen. Dit werk bestudeert een foutenaafhandelsmodel dat leasing verzoent met ambient-georiënteerd programmeren. Het resulterende model noemen we *ambient-oriented leasing*.

We onderzoeken *ambient-oriented leasing* in twee verschillende gedistribueerde programmeermodellen. Ten eerste, integreren we ons model met een gedistribueerd object-georiënteerd programmeermodel door middel van een innovatief referentiemechanisme genaamd *leased object references*. Ten tweede, exploreren we de toepasbaarheid van *ambient-oriented leasing*-abstracties voor tupelruimtes, wat resulteert in een vernieuwende adaptatie van het tupelruimte-model voor MANETs genaamd *TOTAM* (Tuples on the Ambient). We ontwerpen en ontwikkelen beide incarnaties van *ambient-oriented leasing* in *AmbientTalk/M*, een ambient-georiënteerde taal die een geunificeerd metamodel aanbiedt waar zowel taalabstracties als tools op gebouwd kunnen

worden.

Om ondersteuning aan te bieden voor de ontwikkeling van toepassingen die op MANETs draaien bestuderen we tool-ondersteuning onder de vorm van een *ambient-oriented debugger* dewelke helpt om programmeurs een beter inzicht te geven in het dynamisch gedrag van deze toepassingen. We bieden een prototype aan van een online ambient-oriented debugger die geïntegreerd is met de ontwikkelingsomgeving voor AmbientTalk in Eclipse genaamd *REME-D* (Reflective Epidemic MESSage-oriented Debugger).

Tesamen bieden de programmeerabstracties en de ondersteuning voor debugging een innovatie set van werktuigen die de programmeur in staat stelt om partiële fouten het hoofd te bieden in een geanticipeerde en controleerbare manier.