

## SAMENVAT T I N G

Het huidige straatbeeld wordt gekenmerkt door de aanwezigheid van geminiaturiseerde apparaten zoals smartphones en digitale horloges, en binnenkort zelfs digitale brillen. Deze alledaagse objecten zijn meer en meer geïntegreerd met computers, wat ons dichterbij brengt bij Mark Weiser's visie van ubiquitous computing. Deze talrijk aanwezige geminiaturiseerde apparaten kunnen verbonden worden in mobiele ad hoc netwerken door gebruik te maken van standaardtechnologieën zoals WiFi, Bluetooth, NFC, 3G en recent ook 4G. Dergelijke mobiele ad hoc netwerken ontstaan wanneer apparaten spontaan een samenwerking tot stand willen brengen door middel van een draadloos netwerk, zonder hiervoor configuratie of infrastructuur nodig te hebben.

Dit is in tegenstelling tot nomadische netwerken, waar een groep mobiele apparaten een verbinding probeert te behouden met een vaste infrastructuur. Omdat dit type netwerken alomtegenwoordig is (denk bijvoorbeeld aan winkelcentra, luchthavens, etc.), ontstaan rijke toepassingsmogelijkheden.

Wij stellen ons een nieuw soort toepassingen voor, genaamd nomadische toepassingen, die ontwikkeld en uitgevoerd worden op de vaste infrastructuur van de nomadische netwerken en ondertussen complexe interacties met de nabije mobiele apparaten orkestreren. De focus van dergelijke nomadische toepassingen is het orkestreren van de verschillende diensten die worden aangeboden door verbonden servers en door nabije mobiele apparaten. Het orkestreren van diensten leidt tot complex gedrag dat ontstaat uit de interactie tussen deze aanwezige mobiele apparaten en de servers. Orkestratie van diensten in deze context is niet triviaal aangezien nomadische netwerken onderhevig zijn aan volatiele verbindingen. Dit houdt in dat diensten op mobiele apparaten gevoelig zijn voor (tijdelijke) netwerkfalingen. Bovendien zijn dergelijke mobiele apparaten steeds in beweging, wat wil zeggen dat de groep van diensten die georkestreerd wordt dynamisch is tijdens de uitvoering van een nomadische toepassing en dus bijgevolg niet op voorhand bepaald kan worden.

Wij stellen een programmeermodel voor dat de ontwikkeling van nomadische toepassingen vereenvoudigt. Dit programmeermodel unificeert de principes van twee onderzoekszulen, namelijk het workflowparadigma dat zich toespitst op het orkestreren van diensten en het ambient-georiënteerd programmeerparadigma dat gericht is op iii toepassingen die worden uitgevoerd op mobiele apparaten. We definiëren hoogniveau abstracties als workflowpatronen die de orkestratie van diensten in een nomadisch netwerk ondersteunen. Daarvoor dienen we bestaande control flow patronen te herbekijken in de context van nomadische netwerken, waar niet noodzakelijk alle diensten op voorhand gekend zijn en waar diensten (tijdelijk) onbeschikbaar kunnen zijn.

Bovendien biedt het voorgestelde programmeermodel een nieuwe groep van patronen aan die zich richten tot de orkestratie van een dynamische groep van diensten.

Deze verzameling patronen zorgt er voor dat (een deel van) de toepassing uitgevoerd wordt voor alle diensten die aan de groepsbeschrijving voldoen. De volatilititeit die aanwezig is in nomadische netwerken zorgt er voor dat gehele synchronisatie niet altijd mogelijk is. Daarom stellen we meer geavanceerde synchronisatiepatronen voor die de uitvoering van de toepassing controleren op een manier die de individuele uitvoering van een enkel groepslid overstijgt. Omdat nomadische netwerken gebouwd worden op volatiele netwerkverbindingen zijn mechanismes nodig om falingen tijdens de communicatie af te handelen. Het programmeermodel dat wij voorstellen ondersteunt het automatisch afhandelen van falingen om zo te kunnen omgaan met (netwerk) falingen veroorzaakt tijdens het oproepen van een dienst. We definiëren eveneens extra patronen die toepassingsontwikkelaars toelaten om toepassings specifieke compensaties te specificeren voor bepaalde soorten falingen.

We presenteren het design en de implementatie van deze nieuwe patronen in de context van NOW, een nieuwe workflowtaal voor nomadische netwerken. Deze workflowtaal is geïmplementeerd als een abstractielaag bovenop AMBI ENT TALK/2, de state-of-the-art voor coördinatie in mobiele ad hoc netwerken. We voorzien ook een validatie van onze aanpak door de codecomplexiteit van de implementaties van drie nomadische toepassingen in NOW en AMBI ENT TALK/2 te vergelijken. Tenslotte presenteren we ook een validatie van de performantie en schaalbaarheid van NOW.

Iv

## ABSTRACT

Nowadays we are surrounded by all kinds of computing devices, ranging from smartphones to digital watches, and in the near future even digital glasses. Everyday objects are more and more integrated with computers, bringing us closer to Mark Weiser's vision of the ubiquitous computing era. The plethora of these miniaturised devices can be connected in so-called mobile ad hoc networks, using a variety of standard technologies such as WiFi, Bluetooth, NFC, 3G, and recently also 4G. Mobile ad hoc networks emerge from the collaboration of nearby devices using wireless network topologies, without the need of any configuration or infrastructure. This is in contrast to nomadic networks, a special case of mobile ad hoc networks, where mobile devices try to maintain a connection with a fixed infrastructure. As these kinds of networks are omnipresent (for instance, in shopping malls, airports, etc.), an abundance of interesting applications can be supported.

We envision a new type of rich applications, called nomadic applications, which are deployed and executed on the fixed infrastructure of the nomadic network, while communicating with nearby mobile devices. Such a nomadic application is centred around the orchestration of services that are provided by connected servers and nearby mobile devices.

Orchestration of services enables the realisation of meaningful complex behaviour out of the interaction between the aforementioned mobile devices and the servers. The orchestration of these services is not without a challenge, since these nomadic networks are built upon volatile connections. Services residing on mobile devices are exposed to (temporary) network failures.

Additionally, mobile services are always on the move, meaning that the set of services to be orchestrated is dynamic during the execution of a nomadic application and cannot be determined beforehand.

We propose a programming model that facilitates the development of nomadic applications. This programming model unifies the principles of two research pillars, namely the workflow paradigm, which focusses on the orchestration of services, and the ambient-oriented programming paradigm, which is tailored towards applications running on mobile devices. We define high-level abstractions as workflow patterns that allow the orchestration of services in a nomadic network. Therefore, we revise existing control flow patterns in the context of nomadic networks where the different services are not necessarily known beforehand, and can become (temporarily) unavailable.

Additionally, the programming model offers a set of new patterns that allows the orchestration of dynamically changing groups of services. This set of patterns ensures that (part of) the application is executed for all services satisfying the group's description. The inherent volatile connections of the network cause communication partners to disconnect making full synchronisation not always possible. Therefore, we include more advanced synchronisation patterns, that enable control over the execution in a way that transcends the individual process of a single member.

Because nomadic networks are built upon volatile networks, failure handling mechanisms are required to overcome communication faults. The programming model we propose incorporates a default failure handling mechanism in order to overcome (network) failures during service invocations. We define additional patterns that allow application developers to specify compensating actions to handle particular failure events.

We contribute the design and implementation of these novel patterns in the context of NOW, a novel workflow language for nomadic networks. This workflow language is implemented as an extra layer of abstraction on top of AMBI ENT TALK/2, the state-of-the-art programming language for coordination in mobile ad hoc networks.

We also provide a validation of our approach by comparing the code complexity of the implementations of three nomadic applications in NOW and AMBI ENT TALK/2.

Finally, we present a validation of NOW's performance and scalability.

ii