

“Exceptions zijn duur”

INTERVIEW MET ONTWIKKELTEAM VAN EMBEDDED SYSTEMS

Voor de verkoop van kaartjes op de trein en voor de informatieverstrekking aan reizigers was de NMBS (Belgische spoorwegen) op zoek naar een nieuwe handterminal voor de treinbegeleiders. Een team van ingenieurs bouwde zowel de hardware als de software voor een op Windows CE gebaseerde handterminal. Wij spraken met de ontwerpers.

Voor de treinbegeleiders (conducteurs) werd door de Belgische spoorwegen een tender uitgeschreven voor een nieuwe multifunctionele handterminal, met alle wensen en eisen samengevat in een lastenboek. Ralf Geenen, werkzaam bij AE in België (www.ae.be), is vanaf het begin betrokken bij het IBIS-project voor de Belgische Spoorwegen. “Het vorige toestel heet Ivette en dit is de tweede versie, daarom is de naam IBIS (Ivette bis) geworden. [Wij hebben ons als subcontractor van dZine mee ingeschreven op het project.](#)” Om de opdracht binnen te halen moest in een zeer kort tijdsbestek een ‘open-frame’-prototype gebouwd worden, met daarop een aantal testapplicaties. Kris Gielis, werkzaam bij hoofdaannemer dZine (www.dzine.be), vult aan: “Inmiddels zijn er meer dan 3000 exemplaren van het toestel in gebruik door de treinbegeleiders van de NMBS. Het is geen gemakkelijke opdracht gebleken, want de eisen in het lastenboek waren erg hoog. Geen enkele handterminal had op dat ogenblik zoveel verschillende hardwaredevices aan boord. Met een doorsnee Pocket PC of standaard embedded device was het niet mogelijk te voldoen aan de eisen van de NMBS, vandaar zelfbouw.”

Van alles voorzien

Kris Gielis van dZine: “Wij hebben het hele apparaat van scratch af ontworpen. De hardware, de drivers en het besturingssysteem hebben wij voor onze rekening genomen. AE deed de applicatiekant. De enige optie die we hadden voor de ontwikkeling van een dergelijke multifunctionele handterminal was om de elektronica, in essentie het moederbord, maar ook de kunstof behuizing volledig zelf te ontwerpen. Het nieuwe apparaat moest indien gewenst, in staat zijn betalingen met credit cards te kunnen afhandelen. Er zit dus een chipkaartlezer in, maar ook een magneetstriplezer en een GSM/SMS/GPRS-module. Het betreft de MC45 module van Siemens, omdat hiervan ook een GSM-R (GSM-Railway) versie beschikbaar was. Vervolgens zit er een thermische printer in, een scherm met full VGA resolutie, een WiFi-module om in de stations de informatie te kunnen synchroniseren, een barcodelezer en nog een RFID/contactless chipcardreader. Het apparaat moest bovendien in staat zijn om ten minste acht uur autonoom te kunnen werken. De treinbegeleiders laden het apparaat nu in praktijk slechts om de twee à drie dagen op.”

“Aan de applicatiekant gaan we ook slim om met de powerconsumptie”, vult Ralf Geenen aan. “Als er een zware berekening aankomt, dan kan de applicatie dat doorgeven aan het OS zodat de processor meer power krijgt. Om de powerconsumptie verder te beperken gaat het apparaat vrij snel in stand-by, daar moet je met de applicatiesoftware wel goed op inspelen. Er zijn ook allerlei beveiligingen ingebouwd zoals een heartbeat, en als de accu leeg raakt, wordt alles opgeslagen en afgesloten en gaat het apparaat vervolgens in stand-by mode.”

Windows CE

Kris Gielis: “De keuze voor het besturingssysteem was vrij snel gemaakt. De klant had namelijk een grote voorkeur voor Windows CE en wij heb-

ben ons daar bij aangesloten. Het gebruik van de ontwikkelomgeving en het bouwen van images is gemakkelijker dan bij welk ander OS dan ook. We zijn er uiterst tevreden over. In het begin hebben we nog wat opstartproblemen gehad met Windows CE 3.0, maar vanaf 4.2 was het product volwassen en sindsdien loopt alles naar wens.” Ralf Geenen gaat verder: “Op de grote stations kan de treinbegeleider de terminal via WiFi synchroniseren. Elke begeleider heeft zijn eigen handterminal en neemt deze ook mee naar huis. Ze kunnen hun volledige administratie bijhouden en er rapporten en verslagen in schrijven en bovendien hun vakantiedagen opvragen. Elke medewerker is op deze manier ook eenvoudig te bereiken als een collega plotseling ziek wordt. Er zijn er in totaal 3125 geleverd. Hoewel elke treinbegeleider zijn eigen terminal heeft, moeten ze toch van de ene treinbegeleider aan de andere overgedragen kunnen worden. “In België hebben we altijd te maken met twee talen”, vertelt Ralf Geenen. “De applicatie kan eenvoudig omgeschakeld worden van Nederlands naar Frans. Het .NET Compact Framework biedt standaard dezelfde mogelijkheden wat betreft lokalisatie als het standaard .NET Framework. We hebben alle taalafhankelijke zaken in een database gestopt en werken met satellite-assemblies.”

P/Invoke

Ralf gaat verder: “De applicatie is geschreven in managed code met uitzondering van de device-drivers, die zijn grotendeels door dZine in C/C++ geschreven. Je treft daarom in de source-code veel P/Invoke-code. Het liefste wil je zo veel mogelijk in managed code blijven, omdat marshaling van datatypes tussen C en .NET niet eenvoudig is. Op Windows CE was er bijvoorbeeld nog geen printerdriver. De lay-out van de tickets moet redelijk flexibel zijn, want treinkaartjes kunnen veranderen. De NMBS wil via een template de lay-out kunnen bepalen. Om dat mogelijk te maken hebben wij de lay-out van de treinkaartjes in een database-



Afbeelding 1. De IBIS handterminal

structuur opgezet. Nu kan de NMBS zonder in de code te duiken zelf vrij eenvoudig de lay-out van de tickets aanpassen.”

Ook het communiceren met de gsm en met de kaartlezers is vrij complex en gaat allemaal via P/Invoke. “Je hebt geen connection-pooling, de communicatie met de kaartlezers is event-gebaseerd en luistert heel nauwkeurig”, legt Ralf uit. “We hebben behoorlijk aan de threading moeten sleuren. Neem nu bijvoorbeeld de verschillende stadia van een gsm-toestel. Je moet dat in de software goed onder controle hebben. De treinbegeleider kan aan het bellen zijn, terwijl er tegelijk een sms’je binnenkomt. Vertragingen op het spoor worden bijvoorbeeld per sms doorgegeven. Ook voor de kaartlezers waren er geen device-drivers beschikbaar. Er zijn wel standaard interfaces, maar die zijn alleen aanwezig op het ‘grote’ Windows-platform en niet op Windows CE. We hebben dezelfde interfaces geïmplementeerd op Windows CE; mocht Microsoft in de toekomst komen met een device-driver, dan hoeven we de code slechts minimaal aan te passen.”

SQL Server CE

“Dit was ons eerste embedded project”, zegt Ralf Geenen. “We hebben wel veel .NET- en Java-ervaring, maar bij de aanvang van het IBIS-project hadden we geen specifieke embedded-kennis. We zijn een projectgeoriënteerde organisatie en doen hoofdzakelijk projectleiding van grote projecten met duizenden mandagen waarbij wijzelf voornamelijk de analyse, het design en de technische architectuur doen. We zijn gewend om gedistribueerde applicaties te maken, ons design moest voor dit project enorm vereenvoudigd worden. Het is een andere manier van werken. Je hebt heel erg weinig geheugen en wij kwamen al snel in de problemen. Eén oorzaak daarvan was dat SQL Server CE toch wel erg veel geheugen opeiste. Wij hebben de code door Microsoft laten valideren. Het was erg leuk om te horen dat er geen enkele fout in onze code zat. Microsoft heeft wel ervoor gezorgd dat het geheugengebruik van de database-engine van SQL Server CE minder is geworden.”

Ralf: “Eigenlijk was er als eerste voor een Oracle Lite-database gekozen, temeer omdat de NMBS een strategisch contract heeft met Oracle. De geheugen-footprint van Oracle was wel groter dan SQL Server CE, maar Oracle Lite was weer sneller. Dit kwam vooral doordat de indexen allemaal in het geheugen werden bijgehouden. Toch zijn wij uiteindelijk van Oracle Lite afgestapt, omdat het aan de server-kant met de synchronisatie mis ging. Er is besloten naar SQL Server CE over te gaan. Aan onze kant was de data laag goed afgeschermd, alle data-access was geïsoleerd en de overstap was binnen een week gebeurd. Wij hebben daarbij veel commitment van Microsoft gehad om de synchronisatie goed te krijgen. Deze synchronisatie met de backend is niet geheel onbelangrijk, want in het apparaat zijn naast de dienstregeling en andere informatie ook alle financiële transacties opgeslagen. De bank eist bovendien dat die data dubbel opgeslagen worden, zowel in de database als los in een file. Alle bankgegevens en andere gevoelige data worden bovendien beveiligd (encrypted) in het apparaat opgeslagen.”

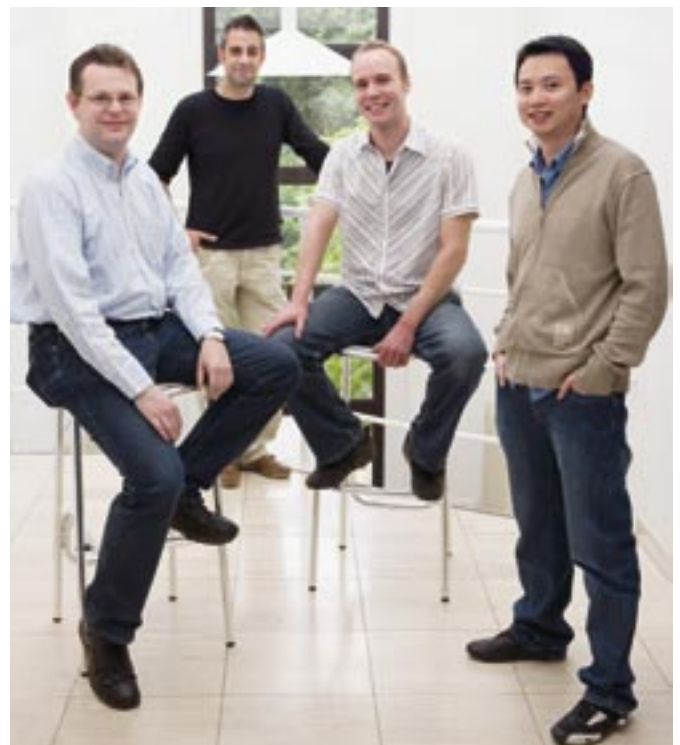
Bediening

Naast de functionaliteit waren er andere technische eisen bijvoorbeeld met betrekking tot schokbestendigheid, ergonomie en het gewicht, waaraan het toestel moest voldoen. Gebruiksvriendelijkheid, hanteerbaarheid en een beperkt gewicht waren erg belangrijk voor de NMBS. Het IBIS-toestel weegt 1,2 kilo en de vormgeving van het toestel is mede door de medewerking van de trienbegeleiders zelf tot stand gekomen. Ralf Geenen: “Ivette – het vorige apparaat - woog een flink stuk meer en was vrij basaal wat betreft functionaliteit en was bovendien karaktergebaseerd. IBIS daarentegen is helemaal grafisch met meer dan 100 schermen en rond 180 tabellen. Het is een grote applicatie, terwijl het geheugen op het toestel zeer beperkt is en dat is niet eenvoudig. De grootste uitdaging was dan ook het beperkte geheugen en het tekenen op het scherm”, zegt Ralf. “De meeste schermen op een standaard Windows CE device zijn vrij eenvoudig. De schermen waar wij mee te maken hebben, zijn vrij complex met veel controls en comboboxen. Als die allemaal gevuld moeten worden met data, loop je snel tegen performanceproblemen aan. Dat is overigens ook zo op het ‘grote’ Windows-platform. Wij hadden wat betreft de schermen weinig speelruimte, want in het lastenboek stond precies

beschreven hoe de schermen er uit moesten zien. Ook de responstijden waren bepaald. Het informeren van een reiziger en een transactie moeten vlot verlopen. In het lastenboek stond ook de eis beschreven voor een Outlook-achtige interface die je met je vinger of een gewone balpen kunt bedienen. Een aparte stylus zou veel te snel zoek raken. We hebben uiteindelijk zelf controls gemaakt, want de standaard controls voldeden niet. Hier hebben onze programmeurs heel veel winst behaald wat betreft performance en geheugengebruik. Ook de ontwikkelaars van dZine hebben hier een bijdrage geleverd.” Kris Gielis: “We hebben een aparte displayprocessor ingezet, in plaats van de display direct aan de processor aan te sluiten zoals in de meeste pocket pc’s. Als gevolg hiervan wordt de processor minder belast door grafische processen, en blijft hij grotendeels beschikbaar voor het echte werk”

Rekening houden met geheugen

“Om een goede en snel werkende applicatie te bouwen voor Windows CE, en eigenlijk voor alle embedded systemen, moet je terdege rekening houden met de geheugencapaciteit. Je kunt niet in het wilde weg programmeren”, maakt Ralf duidelijk. “Exceptions zijn bijvoorbeeld duur. Je moet wel rekening houden met uitzonderingen, exceptions blijven natuurlijk nodig. Maar je moet goed kijken of iets wel een exception is en of je die wel wilt opvangen. Veel developers schrijven automatisch een exception. Als je bijvoorbeeld inlogt en je geeft een foutief wachtwoord, dan schrijven ze er een exception voor. Eigenlijk is dit niet nodig. Want alles wat je kunt verwachten, is geen exception. Wanneer je geen connection krijgt met de database, dan is dat een exception. Als je echter naar een bepaalde naam zoekt in de database en er wordt niets gevonden, dat is dat te verwachten, en dus geen exception. Om daar een exception voor te maken is erg duur. Ook wat betreft databasedesign dien je rekening te houden met het beperkte geheugen. Hoe ver ga je met normalisatie bijvoorbeeld. Je moet goed uitkijken wat je doet want een join over meer tabellen in geheugen is ook erg duur. Eigenlijk is het allemaal common sense. Je dient de code te optimaliseren, te optimaliseren, en nog eens te optimaliseren. Dankzij het IBIS-project hebben we heel veel geleerd, hoofdzakelijk hoe je moet omgaan met heel weinig geheugen. Je bent constant op zoek om dingen slimmer te doen of kleiner te maken. Dat passen we nu ook toe op het .NET-platform. Om een device te laten werken met weinig geheugen is een enorme uitdaging. Ik kan het iedereen aanraden om eens te doen. Want je gaat elke letter van je code twee keer bekijken.”



Afbeelding 2. Ontwikkelaars van Application Engineers. Van links naar rechts: Philippe De Mecheleer, Luc Claesen, Joris Van den storme en Wai Keung Yuen. (Fotografie Hans Oostrum)