

Java speelt een cruciale rol bij Jini, de nieuwe technologie van Sun om apparaten met elkaar te laten samenwerken.

## De geest is uit de fles!

Waarom kan ik een videoband die ik beneden in de videorecorder heb zitten niet boven op de slaapkamer afspelen, terwijl ik daar wel TV kan kijken? Waarom kan mijn TV het beeld van mijn computer niet weer-geven? Waarom komt mijn wekker niet op het idee om meteen het koffiezetapparaat te "booten"? Mensen blinken al sinds het stenen tijdperk uit in het maken van hulpmiddelen. Om taken uit te voeren, te versimpelen en te versnellen. Die hulpmiddelen werken echter verbazingwekkend slecht samen. Alleen als expliciet in bepaalde vormen van samenwerking tussen de apparaten is voorzien, wordt het geheel meer dan de som der delen. Ik ben als consument eigenlijk alleen in staat om apparaten te combineren op manieren die door de leveranciers zijn voorzien. En als samenwerking of integratie al tot de mogelijkheden behoort gaat het vaak om zeer statische en moeilijk tot stand te brengen samenwerkingsverbanden: bijvoorbeeld "TV-videorecorder" of "computer-printer". Sun's "Jini" moet dit gaan veranderen: "Maybe all your toaster needed was someone to talk to....". Achter dit soort "hype-achtige" uitspraken schuilt echter (gelukkig) een interessante en wellicht revolutionaire technologie. Jini is een technologie die het mogelijk maakt om apparaten te combineren op manieren die vooraf niet waren voorzien. Ieder Jini-compatibel apparaat bevat een Java-compliant Virtual Machine die in staat is om over een netwerk met alle andere apparaten in

zijn onmiddellijke omgeving te praten. Een apparaat kan diensten aan het netwerk aanbieden en/of diensten van andere apparaten in het netwerk verlangen. Door een slimme manier te leveren voor het aan elkaar koppelen van vrager en aanbieder kunnen met Jini apparaten worden gecombineerd tot een ad-hoc federatie. Een nieuw apparaat in het netwerk registreert zichzelf bij een centrale "Lookup" service. Bij die aanmelding laat het apparaat ook meteen weten welke diensten het in de aanbieding heeft. Mijn nieuwe Jini-TV laat bijvoorbeeld aan de centrale "Lookup" service (die draait op de huisserver in de meterkast) weten dat het de dienst "Beeldweergave" in de aanbieding heeft. Bij die registratie stuurt de TV echter ook een Java-object (code+data) mee die de communicatie met de TV verzorgt! Nu wil de TV natuurlijk ook geluid weergeven en hij vraagt aan de "Lookup" service welke apparaten die dienst allemaal aanbieden. De "Lookup" service antwoordt dat twee apparaten in mijn netwerk dat kunnen: mijn Jini-compatible versterker en mijn desktop-PC die toevallig twee boxen in de monitor heeft. De TV kan natuurlijk niet kiezen dus vraagt hij aan mij via welk apparaat ik het geluid af wil spelen. Ik kies natuurlijk voor mijn versterker (immers de beste geluidskwaliteit). Desgewenst had mijn TV ook zelf kunnen kiezen op basis van allerlei geluidskwaliteitsattributen die de apparaten bij de centrale "Lookup" service hebben

geregistreerd. Om daadwerkelijk met de versterker te communiceren laadt de TV het Javaservice-object van de versterker en roept methodes in dit object aan. Het service-object is verantwoordelijk voor de communicatie met de versterker om daadwerkelijk geluid te produceren. Java speelt een cruciale rol bij Jini: het Javaplatform is architectuurneutraal, bevat afdoende voorzieningen voor netwerkcommunicatie en, het allerbelangrijkste hier, staat toe dat Java-bytecode-programmatuur over het netwerk wordt verplaatst naar een andere Virtual Machine voor (veilige) verwerking aldaar. Met name deze laatste eigenschap is erg krachtig omdat hiermee het aanroepen van de diensten van een verder onbekend apparaat aanzienlijk wordt vereenvoudigd. Fabrikanten van apparaten moeten natuurlijk wel tot standaardisering komen van welke diensten worden gedefinieerd en hoe ze worden omschreven en aangeroepen. Sun heeft echter al een hoopvol begin hiermee gemaakt door leveranciers als Philips en Sony aan boord van de Jini-trein te krijgen. De geest is uit de fles! Nu nog kijken hoeveel van onze wensen ook door hem worden gehonoreerd.

Jos Visser

is werkzaam als geestenbezweerder bij de Open Solution Providers en is bereikbaar via [josv@osp.nl](mailto:josv@osp.nl).