

De gebeurtenissen in de huidige wereld wisselen elkaar in een razend tempo af en onze software heeft de ondankbare taak te proberen dat tempo bij te houden. Een programmeermodel dat daar goed toe in staat lijkt, heet Event stream processing. Software Release Magazine sprak met Mark Palmer, vice president, Event Stream Processing, Progress Real Time Division.

thema

Event stream processing

Nieuw programmeer-paradigma

Event Stream Processing was tot nu toe een gebied voor pioniers zoals Streambase Systems en Apama. Sinds de overname van Apama door Progress en de integratie daarvan in het Progress productaanbod, lijkt het echter met grote stappen een grote toekomst tegemoet te gaan.

Wat is ESP precies?

Palmer: 'Het wordt vaak *complex event processing* genoemd. Het is tegenwoordig een academisch vakgebied en beschrijft een manier van programmeren die van doen heeft met tijdelijke en complexe sequenties van

De taal is een voorbeeld van het hoge niveau tool dat we hebben en ontwikkelaars gebruiken deze tool om er deze casual constraints in uit te drukken. Neem een RFID-systeem, waarbij je verschillende staten doorloopt. Ik lees een pallet en identificeer een levering, krijg een factuur, valideer die pallet en handel fouten af. Elk van deze regels heeft een venster waar je een aantal regels code hebt. Er zit dus een beetje SUDO-code bij, maar de ontwikkelaar zal zo ongeveer alleen de regel zien. Deze regels code zijn gecompliceerd. Wanneer je er één patroon mee beschrijft gaat het nog wel. Wanneer je echter heel veel moet schrijven, wordt het bijna onmogelijk, dus daarom gebruik je een tool. *Event stream processing* heeft dus deze manier van programmeren nodig.'

Dit is dus een nieuw programmeer-paradigma.

Palmer: 'Heel nieuw. Het maakt het mogelijk dit soort problemen veel eenvoudiger op te lossen, maar aan de andere kant zou het voor een programmeur die gewend is op een ander manier te denken, moeilijk kunnen zijn.

Het voorbeeld dat ik gegeven heb, is geschreven in wat wij *monitor script* noemen, een event-programmeertaal. (Een op het eerste gezicht zeer eenvoudige taal die dicht bij natuurlijke talen staat, DdM.) Er zijn verschillende programmeer-interfaces. Je kunt dus *monitor script* gebruiken, Java, .NET-talen en daarnaast is er ondersteuning voor de Microsoft *real time data browsing*. Voor Java-programmeurs zijn dit gewoon objecten, waarmee je zegt: wanneer er een event optreedt voor dit object, doe dan dit of dat. Het is een heel andere programmeerstijl, maar voor Java-programmeurs is het zeer gemakkelijk op te pikken. Het is echter een andere metafoor, het is een event driven metafoor voor gebeur-

Het is een heel andere programmeer-stijl, maar voor Java-programmeurs is het zeer gemakkelijk op te pikken

gebeurtenissen (events) en over de mogelijkheid deze patronen uit te drukken in een taal. In een doorsnee systeem kun je honderdduizenden *event streams* hebben. In het voorbeeld dat ik zo zal geven is maar een klein deel vertegenwoordigd, maar in een trading systeem zul je er duizenden hebben. Je kunt het beschouwen als een regel, maar het verschilt erg van een traditionele *rules engine*. Je kunt namelijk predikaten aangeven zoals: als de prijs van de afzonderlijke aandelen Microsoft gedeeld door het aantal aandelen gedurende vijftien minuten verandert, én gebeurtenis x treedt op, let dan op andere events. Bijvoorbeeld: als Microsoft verandert en de SMP gaat omhoog en óf Microsoft stijgt óf daalt binnen een periode van twee minuten, dan volgt daaruit: koop Microsoft en verkoop HP.

tenissen die je tot nu toe anders zou formuleren. Je hebt eventjes nodig om eraan te wennen maar de meeste ontwikkelaars van bedrijven die op de beurs handelen, pakken het binnen een paar dagen op.'

Hoe ervaren mensen het die al bezig zijn dit soort problemen op te lossen in conventionele talen?

Palmer: 'Voor hen is het heel bevrijdend, want tot nu toe was het voor hen heel moeilijk deze problemen op te lossen.'

Wanneer je zonder deze methode werkt, krijg je waarschijnlijk SQL-statements die enorm complex zijn.

Palmer: 'In SQL werkt het helemaal niet, want de requirements die je hier hebt, zijn meestal heel grote aantallen, dus high performance query's, maar ook een hele lage latency (traagheid). De tijd die verstrijkt tussen het moment dat zo'n prijs van een aandeel verandert en de andere dingen, als ze gebeuren, is heel kort. Daarom noemen we het *real time data management*. Het real time-aspect, in feite het werken binnen een zeer klein tijdvenster, is het grootste deel van het probleem. Al sinds 1985 wordt hier onderzoek naar gedaan, waarbij data opgeslagen worden in een database om er vervolgens query's over te laten draaien. Het gaat dan juist om heel grote aantallen data, het opslaan van de data, het indexeren en beschikbaar maken, zodat je kunt query's. De manier waarop mensen dit hebben proberen te doen, voegt seconden tot zelfs minuten toe na de verwerkingstijd, afhankelijk van de hoeveelheid data die binnenkomen. Die traagheid is onacceptabel. Je kunt een vrachtwagenchauffeur niet zeggen: blijf nog maar even wachten in het distributiecentrum tot het systeem het allemaal uitgerekend heeft.'

Hoe werkt het technisch, wat zit er allemaal achter?

Palmer: 'In het hart zit de event manager. De event server is het hart ervan en de taal die ik net beschreef is in die event server gedeployed. Je kunt het zien als een applicatieserver maar dan voor events. De regels worden hier gedeployed, de event server monitort de events en let op de patronen. Het gaat zeer sophisticated. Je kunt het je voorstellen als een database server, maar dan voor real time data, een event streamserver. Het kijkt continu naar de events, kijkt naar de patronen en de query's terwijl de events plaatsvinden.'

Ik neem aan dat het geheim ervan is dat het werkt met redundancy, dat dingen die niet belangrijk zijn niet verwerkt worden en misschien zelfs niet eens opgeslagen.

Palmer: 'Ja, dat klopt, het werkt in feite met een gigantische kaart van event patterns. Als er een event optreedt dat niet eens gedeeltelijk aan één van de te beoordelen events beantwoordt, dan let de event manager er niet op. Bijvoorbeeld, als de SMP een half procent



omhoog gaat, maar het aandeel Microsoft verandert niet in prijs.'

Slaat het de gebeurtenis dan ook niet op?

Palmer: 'Een ander gedeelte van ons product bestaat uit de event store. Die kán en zal vaak alles opslaan. Er zijn namelijk ook vaak query's in de trant van: wanneer de prijs van het aandeel Microsoft omhoog gaat met twee procent en er volgt nog iets op én de trend van Microsoft in de laatste vijf minuten was omlaag, doe dan x. Het is dus nodig om ook naar de geschiedenis van data te kijken, en daarin voorziet de event store. Andere componenten zijn dashboards, allerlei soorten technologie om een grafische view te maken van wat er gebeurt. Daarnaast zijn er adapters, bijvoorbeeld voor de verschillende financiële dataleveranciers, zoals Reuters. Een adapter helpt ook om bijvoorbeeld een beslissing in een daadwerkelijke aankoop om te zetten. Ons product bestaat dus uit dashboard-technologie, uit de *event store* - de event server is het hart -, uit de development omgeving, en dan zijn er nog de adapters naar de event sources.'

Zoals ik het nu zie, is het mogelijk om data te verwerken, op een schaal die zelfs met een extreem grote en dure BI-omgeving moeilijk zou worden.

Palmer: 'Zeg maar rustig onmogelijk, want dat is een statisch model en het is te traag. BI is volledig gebaseerd op hetzelfde model: pak de data, sla ze op, indexeer ze en bevraag ze. Voor elke lage traagheids-omgeving zoals algoritmische trading is het onmogelijk.'

Waarom probeert u niet een marktaandeel te winnen in dat deel van BI?

Palmer: 'Gedeeltelijk doen we dat ook. Bij het tunen van algoritmen ontdek je weer nieuwe patronen. Wanneer je dat eenmaal gedaan hebt kun je ertoe overgaan, vragen te stellen in de geschiedenis van je data. Dat is iets waarnaar we onze omgeving uitbreiden. BI is

heel goed in het doen van dat soort dingen voor statische data, maar wij zijn heel goed in het doen van die dingen voor stromen van data. Als je dus wilt kijken naar de geschiedenis die je hebt en je ziet: dit zijn de condities die deze mogelijkheid hebben doen ontstaan, dan is dat het soort vragen dat we zouden willen stellen. Het gaat niet alleen om het rekenen met de data die er al zijn, het is meer: laat ze terug stromen en ga op zoek naar die patronen. Dat is een variant op traditionele BI.'

Zou het dan geen ideale toevoeging zijn op BI?

Palmer: 'Ja, absoluut.'

Dré de Man

PATCHES Patches PATCHES Patches PATCHES Patches PATCHES

Artikelen over onderwerpen als software-ontwikkeling, Java, UML, eXtreme Programming en nog veel meer vindt u in het Online Archief van Array Publications. Vaktijdschriften als Software Release, Java Magazine, Database Magazine en ons Oracle vakblad Optimize hebben hun artikelenarchief online gezet. Dankzij de heldere zoekstructuur vindt u snel wat u zoekt op www.release.nl.

Progress lanceert OpenEdge 10.1

De Progress OpenEdge Divisie, onderdeel van de Progress Software Corporation, introduceert de nieuwe Progress OpenEdge 10.1 productlijn. De introductie vond plaats op Progress Technology World in Lissabon. In deze nieuwste versie van OpenEdge is een nieuwe op Eclipse gebaseerde geïntegreerde ontwikkelingsomgeving (IDE) inbegrepen, maar ook accounting services. Dit is ontwikkeld om de controlemechanismen van de financiële dienstverlening, gezondheidszorg, en andere bedrijfstakken die te maken hebben met boekhoudcontroles en nalevingvereisten, aan te pakken. Het controleren van de boekhouding is de eerste in een reeks van toekomstige OpenEdge functies die het makkelijker zullen maken voor partners en klanten om zakelijke applicaties te bouwen die voldoen aan deze verordeningen.

"Progress partners zijn blij dat zij de experts zijn in deze

bedrijfstak, en we maken het makkelijker voor hen om applicaties te ontwikkelen die die expertise weerspiegelen," zegt Niel Powers, Vice President Products van de OpenEdge Divisie. "OpenEdge 10.1 lost de belofte van Progress in om de complicaties uit het ontwikkelen van geavanceerde, data driven, schaalbare applicaties te halen die gebouwd zijn op een servicegeoriënteerde architectuur - Service Oriented Architecture (SOA). Door betere componentendiensten, verbeterde taalmogelijkheden, en nieuwe tools te leveren door een standaard open-source omgeving te gebruiken, kunnen onze applicatiepartners zich richten op de functionaliteit van de applicatie wat hun core business is.

De nieuwe ontwikkelingsomgeving is gebouwd op het Eclipse framework. Dit nieuwe aanbod, OpenEdge Architect, is op maat gemaakt om applicaties te bouwen door een architectonische, zakelijk-logische aanpak te hanteren. Deze aanpak bevordert de

mogelijkheid van een ontwikkelaar om servicegerichte applicaties samen te stellen uit een catalogus van componenten en diensten, die het hen mogelijk maken sneller aan zakelijke vereisten te voldoen. Er is ingebouwde connectivity beschikbaar naar bericht- en integratiecomponenten vanuit Sonic Software, een ander bedrijfsonderdeel van Progress Software Corporation. De combinatie van het OpenEdge platform met de Sonic SOA Suite maakt het voor ontwikkelaars mogelijk om de OpenEdge-applicaties eenvoudig te integreren met heterogene omgevingen, ofwel door het bedrijf heen of tussen bedrijven onderling. Boekhoudkundige diensten op het niveau van de applicatie en de database worden als te configureren diensten 'out of the box' geleverd, hetgeen ontwikkelaars aanzienlijke inspanningen bespaart. De boekhoudkundige diensten breiden het concept van veiligheid en audit-integriteit uit met mechanismen die verzekeren dat het

boekhoudkundige traject van transacties zelf veilig zijn en bestand zijn tegen fraude. Nieuwe failover/failback-mogelijkheden, gecombineerd met meer online administratie mogelijkheden en het stroomlijnen van routine beheerstaken, staan meer controle op de gegevensbeheer-omgeving toe en verzekeren grotere beschikbaarheid van systemen en gegevens. Nieuwe taalconstructies en mechanismen maken het eenvoudiger om herbruikbare service-componenten te ontwerpen en te implementeren voor een snellere constructie van servicegerichte applicaties.