

The Big BAM

METEN IS WETEN

Vincent Scheel en Rémon ter Haar

Business activity monitoring (BAM) biedt waardevolle inzichten in bedrijfsprocessen. BAM is een instrument om bedrijfsprocessen te meten. Dit artikel maakt duidelijk hoe je de voordelen van BAM gemakkelijk kunt toepassen.

Elektronische data uitwisseling wordt voor veel bedrijven steeds belangrijker. Daarnaast worden de bedrijfsprocessen in moderne informatiesystemen steeds complexer. Tijdens het uitwisselen van informatie met partners kunnen diverse fouten optreden, zowel functioneel als technisch. In veel gevallen wordt bij het monitoren van informatiesystemen voornamelijk technische informatie verzameld en blijft functionele informatie achterwege. De functionele informatie verschaft het management waardevolle informatie, die gebruikt kan worden voor het verbeteren van het proces, het nemen van beslissingen of voor het controleren van een Service Level Agreement (SLA). Daarnaast kan functionele informatie gebruikt worden bij het inzichtelijk maken van doorlooptijden, het meten van gemiddelde orderprijzen of bij het traceren van orderinformatie. Aan de hand van dit soort 'Key Performance Indicators' (KPI's) is het mogelijk om de bedrijfsprocessen in de gaten te houden en deze, indien nodig, bij te sturen. De verzamelde informatie kan ook gebruikt worden in een Business Intelligence (BI) oplossing. BizTalk Server ondersteunt Business Activity Monitoring (BAM) sinds de 2004 release en BAM is in de 2006 release nog verder uitgebreid. BAM is een veelal onderbelicht onderdeel van BizTalk. BAM is een term uit de Business Process Management (BPM) hoek, terwijl BizTalk meer met techniek en softwareontwikkeling wordt geassocieerd. Veel BizTalk en .Net ontwikkelaars geven aan zelden of nooit de BAM-faciliteiten van BizTalk te gebruiken. Daar komt bij dat veel BPM-consultants eveneens onbekend zijn met de BAM-mogelijkheden die BizTalk te bieden

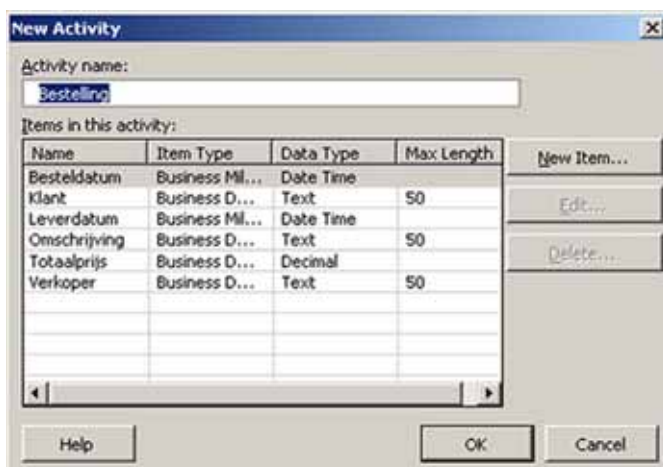
heeft. Het feit dat Microsoft BAM alleen als onderdeel van de BizTalk licentie en niet als losstaand product aanbiedt, heeft hier natuurlijk ook mee te maken.

Dit artikel laat ontwikkelaars zien dat BAM eenvoudig, achteraf en volledig los van een BizTalk-oplossing toegepast kan worden en direct waardevolle informatie voor zowel de business- als de IT-afdeling oplevert. Daarnaast willen we laten zien dat niet alleen BizTalk-ontwikkelaars, maar ook .Net-ontwikkelaars profijt van BAM kunnen hebben dankzij de BAM API. Dit artikel gaat voornamelijk over het praktische gebruik van BAM en gaat onder andere in op de BAM-Excel add-in, de Tracking Profile Editor, de eventstreams, de BAM portal en de BAM webservices. De volgorde van het artikel komt overeen met de volgorde waarin de verschillende onderdelen van BAM binnen BizTalk worden gebruikt.

BAM Excel Add-in

Met behulp van de BAM Excel Add-in bepaal je welke gegevens je wilt monitoren. Om deze gegevens vast te leggen, definieer je activiteiten in een Excel sheet. Tijdens de installatie van BizTalk Server 2006 inclusief BAM zal de Add-in beschikbaar komen. Deze is te vinden in %PROGRAMFILES%\Microsoft BizTalk Server 2006\Tracking\.

Een BAM activity is een business entiteit of proces waarvan gegevens bewaard moeten worden. Een activity wordt omgezet naar een SQL-tabel in de BAM-database, waarbij de activity items de kolommen vormen. Een voorbeeld van een activity is een bestelling uit een bestelproces (zie afbeelding 1). Activity items zijn er in twee soorten: Business Data attributen, zoals de totaalprijs en de naam van de klant, en Business Milestones, zoals de besteldatum en de uiteindelijke leverdatum van de bestelling. Deze milestones zijn belangrijke momenten in het proces die bewaard moeten worden. Deze momenten worden als DateTime in een tabel opgeslagen. De activity items worden gerepresenteerd door de velden in de betreffende activity tabel. Als alle te monitoren gegevens zijn bepaald, kan begonnen worden met het definiëren van de verschillende activity views. Een activity view is te vergelijken met een SQL view. Vaak zul je verscheidene views definiëren voor verschillende doelgroepen. Met behulp van de views kunnen gebruikers de voor hen relevante en geautoriseerde gegevens zien. Voor applicatiebeheerders bijvoorbeeld zijn views met doorlooptijden belangrijk en voor bedrijfsanalisten views met realtime ver-



AFBEELDING 1: DEFINIËREN VAN EEN ACTIVITY MET DE BAM EXCEL ADD-IN.

koopresultaten. Met views kan bepaald worden wie toegang heeft tot welke informatie. Hiermee kan bijvoorbeeld voorkomen worden dat een technisch beheerder inzicht in de verkoopresultaten krijgt. Binnen een view worden measures en dimensions gedefinieerd. Met een measure (geaggregeerde meetwaarde) wordt datgene bedoeld wat je specifiek wilt meten of vastleggen, zoals de gemiddelde duur van het afhandelen van een proces. Een dimension verwijst naar de context waarbinnen je een measure wilt bekijken. Zo kun je bijvoorbeeld de gemiddelde duur van het afhandelen van je orderproces groeperen per salesmanager, maar wellicht ook per klant of jaartal. 'Salesmanager', 'klant' en 'jaartal' zijn dan dimensions van de measure 'gemiddelde duur orderproces'.

Als alle activiteiten, views, measures en dimensions gedefinieerd zijn, moeten deze geïmporteerd worden naar de BAM-database met de BAM utility BM.EXE. Deze is te vinden in dezelfde directory als Excel Add-in. Vanaf de commandline moet het volgende commando worden gegeven:

```
bm.exe deploy-all -DefinitionFile:MyObservationModel.xls
```

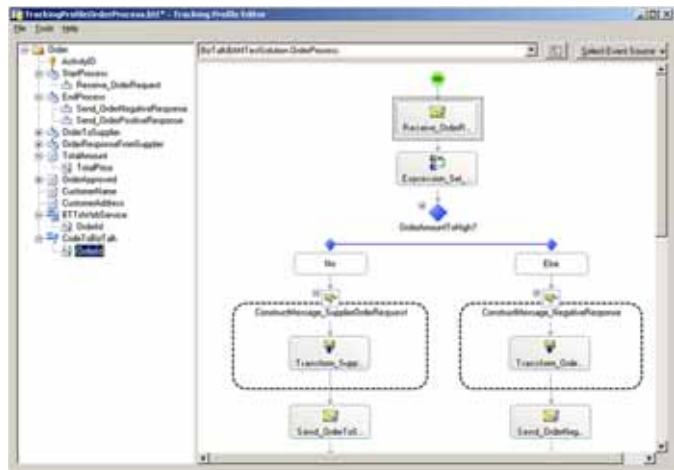
In afbeelding 2 is te zien dat er in de BAMPrimaryImport database een aantal tabellen is aangemaakt dat betrekking heeft op de activity 'bestelling' uit het observation model.

De geaggregeerde gegevens kunnen op twee manieren tot stand komen: via realtime aggregation (RTA) of via scheduled aggregation. Bij RTA wordt de data door een SQL-trigger geaggregeerd in dezelfde transactie als waarin het BAM event wordt geïmporteerd. Hierdoor zijn de gegevens nagenoeg realtime. De gegevens worden bij RTA opgeslagen in SQL-tabellen. RTA wordt vaak gebruikt om KPI's te monitoren. Voor RTA geldt dat er minder dimensions mogelijk zijn dan bij scheduled aggregation en dat slechts data over een relatief korte periode verwerkt kan worden om de performance acceptabel te houden.

Bij scheduled aggregation worden de gegevens op vaste tijden door een DTS package of SQL Server Integration Services (in geval van SQL2000, resp. SQL2005) opgeslagen in een OLAP-kubus. Vanwege het tijdsinterval zijn de gegevens niet real-time. Voor scheduled aggregation heb je een Microsoft SQL Server Analysis Services licentie en een goed verversingsinterval nodig. Scheduled aggregation wordt vaak gebruikt om trend-analyses over een langere termijn uit te voeren.

Tracking Profile Editor

De BizTalk Tracking Profile Editor (TPE) is een tool die je in staat stelt om op verschillende plaatsen een thermometer in een



AFBEELDING 3: TRACKING PROFILE EDITOR.

orchestration (een BizTalk proces) te steken. Data die via receive ports BizTalk binnenkomt, maar ook gebeurtenissen in orchestrations, kunnen met de TPE gekoppeld worden aan de activity items uit het observation model. Dit gaat erg eenvoudig met drag and drop, waarbij links op je scherm de activity items zichtbaar zijn en aan de rechterkant een XSD-schema van een bericht of een orchestration zichtbaar is (zie afbeelding 3). Orchestration shapes kun je aan milestones koppelen. Hierdoor wordt het tijdstip vastgelegd waarop de shape geraakt (bereikt) wordt. De duur van het totale proces kan eenvoudig bepaald worden door de milestone aan het begin van het proces te koppelen aan de receive shapes en aan het eind aan de send shapes van de orchestration. Deze informatie kan gebruikt worden om te controleren of aan de SLAs wordt voldaan. Het grote voordeel van het gebruik van de TPE is dat bestaande orchestrations niet aangepast hoeven te worden.

Eventstreams

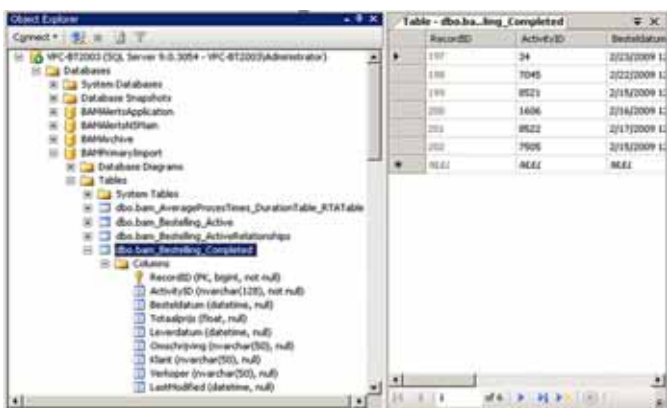
Een van de interessante dingen die de BAM API te bieden heeft, is de mogelijkheid om buiten BizTalk data in de BAM-database te verzamelen of bij te werken, bijvoorbeeld het vastleggen van informatie uit een .Net-applicatie of webservice. Hiertoe zijn eventstreams beschikbaar. Eventstreams stellen je in staat om de BAM-database te vullen via zelf geschreven code.

De volgende eventstreams zijn beschikbaar:

- + OrchestrationEventStream (OES);
- + MessageEventStream (MES);
- + BufferedEventStream (BES);
- + DirectEventStream (DES).

De eerste drie werken asynchroon. Het bijwerken van activiteiten met deze streams loopt via de BizTalk MessageBox. De MES maakt hierbij gebruik van de pipeline transactie en de OES loopt via de orchestration transactie. Hierdoor lift BAM mee op de bestaande SQL roundtrips, waardoor dit bijna geen impact heeft op de performance. De BES en de DES kunnen ook buiten BizTalk-oplossingen worden gebruikt, bijvoorbeeld in webservices of Windows-applicaties.

Om dit te illustreren, gebruiken we de activity uit afbeelding 1, een bestelling die geïdentificeerd wordt door een bestelnummer. In codevoorbeeld 1 wordt de omschrijving van de bestelling op 'Verwerkt' gezet. Zo kan de business analist later de verwerkte bestelling gemakkelijk opzoeken. Via een DES wordt een verbinding



AFBEELDING 2: RESULTAAT DEPLOYEN OBSERVATION MODEL MET BM.EXE

opgezet naar de BAMPrimaryImport database en vindt er een update actie plaats (UpdateActivity), waarna de activity wordt beëindigd (EndActivity). Zo wordt op eenvoudige wijze informatie die enkel buiten BizTalk bekend is, toch bij dezelfde activity in de BAM database bewaard. Wanneer binnen een organisatie een BizTalk Server licentie aanwezig is, kunnen alle .Net applicaties van de BAM-functionaliteit gebruik maken – zelfs als de overige functionaliteit van BizTalk niet wordt gebruikt.

```
string activityName           = "Bestelling";
string activityInstance      = bestelling.BestelNummer;
string activityItem          = "Omschrijving";
string activityItemValue    = "Verwerkt";

DirectEventStream des = new DirectEventStream
("Integrated Security=SSPI;Data Source=.;
Initial Catalog=BAMPrimaryImport", 1);

des.UpdateActivity(activityName, activityInstance,
activityItem, activityItemValue);
...
des.EndActivity(activityName, activityInstance);
```

CODEVOORBEELD 1: UPDATEN VAN EEN BAM ACTIVITY MET C# CODE.

Continuation

Het monitoren van gegevens kan soms lastig zijn. Denk aan de volgende situaties:

- + Een proces kan in een ander systeem beginnen dan waar het eindigt. Dit maakt het lastig om de totale doorlooptijd van het proces te meten.
- + Entiteiten hebben niet altijd in elk systeem dezelfde identifier. Toch moeten ze aan dezelfde activity informatie toevoegen.
- + Communicatie kan asynchroon verlopen. Activity updates kunnen hierdoor in een verkeerde volgorde verwerkt worden. De data in de activity is dan onjuist.
- + Soms is informatie niet in elk systeem aanwezig en moet een ander systeem deze informatie later aan de activity toevoegen.

In al deze situaties moet je Continuation gebruiken. Continuation zorgt voor het aan elkaar relateren van gegevens over systemen heen. Continuation kan door een eventstream vanuit code en vanuit de TPE gebruikt worden. Hierbij wordt onderscheid gemaakt tussen twee verschillende richtingen die in de volgende paragrafen worden uitgelegd.

- + activity starten vanuit code en aanvullen vanuit de TPE
- + activity starten vanuit de TPE en aanvullen vanuit code

Continuation van code naar de TPE

Continuation van code naar de TPE wordt voornamelijk gebruikt in situaties waarbij de activity buiten een orchestration begint en in een orchestration verder aangevuld moet worden. Een voorbeeld is als je vanuit een willekeurige applicatie een als webservice gepubliceerde orchestration aanroept.

Om continuation vanuit code mogelijk te maken, dient vóór de call naar de EndActivity() methode, de EnableContinuation() methode aangeroepen te worden (zie code voorbeeld 2). Geef aan deze methode de naam en de identifier van de activity mee, en ook een continuation identifier die later in de TPE beschikbaar is. In de TPE dient een ContinuationID aangemaakt te worden, die gevuld wordt met de continuation identifier waarmee de continuation is aangezet. Met deze ContinuationID weet de TPE bij welke activity de aangeboden informatie hoort. De continuation identifier bestaat uit twee delen: de naam van de aangemaakte ContinuationID in de TPE plus de waarde van de identifier. In

het voorbeeld (afbeelding 3 en code voorbeeld 2) zal de continuation identifier bij OrderID 1234 de waarde CodeToBizTalk1234 hebben. Op afbeelding 3 zien we een ContinuationID in de TPE gekoppeld aan het OrderId (in de afbeelding geselecteerd) uit de eerste receive shape.

```
string OrderId = "1234";
string OrderActivityName = "Order";
string CustomerAddressActivityItemName = "CustomerAddress";
string ContinuationID = "CodeToBizTalk";

DirectEventStream des = new DirectEventStream("Integrated-
Security=SSPI;Data Source=localhost;Initial Catalog=BAMPrimary-
Import", 1);

des.BeginActivity(OrderActivityName, OrderId);

des.UpdateActivity(OrderActivityName, OrderId, CustomerAddress-
ActivityItemName, textBoxAddress.Text);

des.EnableContinuation(OrderActivityName, OrderId, ContinuationID
+ OrderId);

des.EndActivity(OrderActivityName, OrderId);
```

CODE VOORBEELD 2: CONTINUATION VANUIT CODE NAAR DE TPE.

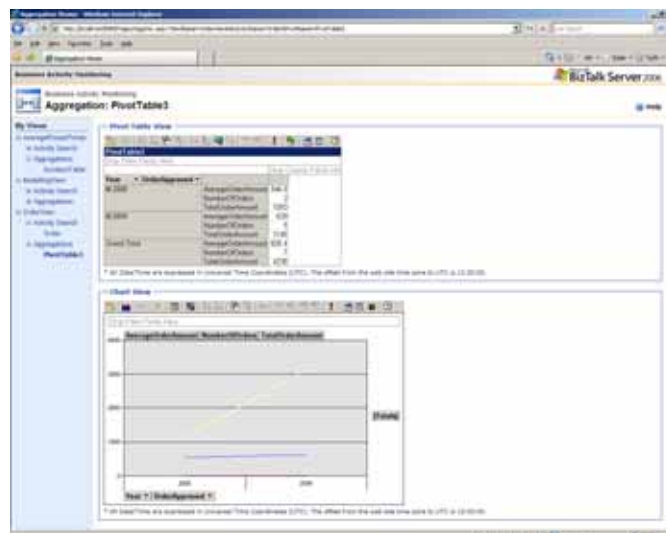
Continuation van TPE naar code

Deze richting van continuation wordt toegepast wanneer er na een orchestration nog meer informatie vastgelegd moet worden, bijvoorbeeld bij het aanroepen van een webservice vanuit een orchestration.

Om deze richting van continuation mogelijk te maken dient in de TPE een continuation folder, in plaats van een ContinuationID, aangemaakt te worden. Aan de continuation folder dient een veld gekoppeld te worden met een identifier die later ook in de code bekend is. In de code worden enkel de UpdateActivity() en EndActivity() methoden aangeroepen. Bij deze methoden wordt de continuation identifier in plaats van het ActivityId meegegeven. De continuation identifier bestaat ook hier uit twee delen: de naam van de aangemaakte continuation folder in de TPE + de waarde van de identifier. Let op dat niet de BeginActivity() methode wordt aangeroepen, want de activity is al gestart door de TPE.

BAM Portal

De BAM portal biedt gebruikers de mogelijkheid real-time inzicht te krijgen in hun bedrijfsprocessen via een webbrowser (zie



AFBEELDING 4: DE BAM PORTAL.

afbeelding 4). De portal is een onderdeel van BAM dat vanaf de BizTalk Server 2006 release beschikbaar is gekomen. Binnen de portal is het mogelijk queries te definiëren, die je in staat stellen om antwoord te krijgen op vragen als: 'Hoeveel orders zijn er de afgelopen maanden verwerkt?' en 'Bij hoeveel orders duurde de afhandeling langer dan de afgesproken levertijd van een week?'

De portal maakt het ook mogelijk om zogenaamde BAM Alerts te definiëren. Hiermee kun je bij het optreden van bepaalde gebeurtenissen in het bedrijfsproces een notificatie sturen. Voorbeelden zijn: het overschrijden van de maximale leveringstermijn uit de SLA, of een bestelling die geplaatst is door een klant met een dubieus betalingsverleden. Deze alerts kunnen ook vanuit code ingesteld worden met behulp van de BizTalk BAM API. De eerder besproken RTA-gegevens kun je direct in de portal bekijken met een office webcontrol. Deze toont per gedefinieerde view een tabel met een grafiek van de beschikbare data. Hiermee is 'out of the box' een gebruiksvriendelijke weergave van de data beschikbaar. In afbeelding 4 is een voorbeeld te zien van een view waarin het aantal orders en de gemiddelde ordertotalen zijn gegroepeerd op jaartal.

BAM Webservices

Alle functionaliteit van de BAM portal is ook beschikbaar via de BAM webservices. Zo kun je queries definiëren en uitvoeren op de BAM database en de resultaten volgens de huisstijl tonen in een eigen (web)applicatie (zie codevoorbeeld 3). Om dit voor elkaar te krijgen, moet er een webreference gelegd worden naar <http://<server>/BAM/BAMQueryService/BamQueryService.asmx>. Vervolgens kun je de webmethod `GetInstanceData()` aanroepen met een query die je zelf definieert.

```
// WS proxy namespace
using BQS = WindowsApplication.BAMQueryService;
...
BQS.BamQueryService bqs = new BQS.BamQueryService();
bqs.Credentials = System.Net.CredentialCache.DefaultCredentials;

BQS.InstanceQuery query = new BQS.InstanceQuery();
query.SelectClauses = new string[]
{ "Omschrijving", "SnelheidAfhandeling" };

BQS.ArrayOfWhereClausesFilterFilter whereClauseFilter = new BQS.
ArrayOfWhereClausesFilterFilter();
whereClauseFilter.Field = "SnelheidAfhandeling";
whereClauseFilter.Condition = "=";
whereClauseFilter.Logic = "And";
whereClauseFilter.Value = "Traag";

query.WhereClauses = new BQS.ArrayOfWhereClausesFilterFilter[]
{ whereClauseFilter };

BQS.Column[][] result = bqs.GetInstanceData("BestellingView",
"Bestelling", query, 10);
```

CODEVOORBEELD 3: UITVOEREN VAN QUERY OP ACTIVITY UIT BAMPRIMARYIMPORT DATABASE.

Met een webreference naar <http://<server>/BAM/BAMManagementService/BamManagementService.asmx> kunnen bijvoorbeeld beheer op alerts en subscriptions op alerts via code worden benaderd. In codevoorbeeld 4 zie je hoe met behulp van deze services een bestaande BAM alert genaamd "LangLopendeAfhandelingen" tijdelijk uitgeschakeld kan worden.

```
// WS proxy namespace
using BMS = WindowsApplication.BAMManagementService;
```

```
...
BMS.BamManagementService bms = new BMS.BamManagementService();
bms.Credentials = System.Net.CredentialCache.DefaultCredentials;

BMS.Alert alert = bms.GetAlert("BestellingView",
"LangLopendeAfhandelingen");
alert.AlertInstance.Enabled = false;

bms.UpdateAlert(alert);
```

CODEVOORBEELD 4: CONFIGUREREN VAN ALERTS VIA DE BAMMANAGEMENT-SERVICE.

Toegevoegde waarde met weinig inspanning

Business activity monitoring biedt waardevolle inzichten in bedrijfsprocessen. Het introduceren van BAM vergt geen aanpassingen aan bestaande BizTalk oplossingen en is in feite niet ingewikkeld. Het samenstellen van het observation model kan door een business analyst in Excel gedaan worden, waarna een ontwikkelaar dit model gemakkelijk met de TPE of met code aan de BizTalk/. Net oplossing kan koppelen. Om de verzamelde gegevens inzichtelijk te maken, kan gebruik worden gemaakt van de meegeleverde BAM portal. Doordat de data in standaard SQL-tabellen of OLAP-kubussen wordt opgeslagen, kun je deze ook gemakkelijk in bijvoorbeeld Microsoft Office PerformancePoint Server of SQL Server Reporting Services gebruiken.

Inmiddels is BizTalk Server 2009 beschikbaar. Wat betreft BAM is er functioneel gezien niets nieuws te melden. Voor wie met BizTalk Server 2009 met BAM aan de slag wil gaan, moet er rekening mee houden dat SQL Server Notification Services (nodig voor BAM alerts) geen onderdeel uitmaakt van SQL Server 2008. Op het moment van schrijven is SQL Server 2008 de database server die minimaal vereist wordt door BizTalk Server 2009. Onduidelijk is of BizTalk Server 2009 ook SQL Server 2005 zal ondersteunen. Microsoft stelt wel de SQL Server 2005 Notification Services Components Package RC1 beschikbaar, die geschikt is voor zowel SQL Server 2005 als SQL Server 2008 databases. BAM maakt deel uit van alle BizTalk Server licenties en het is, naar onze mening, zonde om het niet te gebruiken. Bekijk de huidige BizTalk oplossingen en pas BAM waar mogelijk toe. Binnen een korte tijd kan waardevolle informatie zowel over het proces als uit het proces zelf gehaald worden, zonder die processen zelf te hoeven aanpassen. Een duidelijk overzicht van doorlooptijden kan voor het monitoren van een SLA erg waardevol zijn en is met behulp van BAM snel te realiseren.

Links

Keith Lim's BizTalk BAM Blog: <http://blogs.msdn.com/keithlim/>

Professional BizTalk Server 2006: <http://www.wrox.com/WileyCDA/WroxTitle/Professional-BizTalk-Server-2006.productCd-0470046422.html>

Artikel Carlo Poli 'Inzicht in bedrijfsprocessen met BizTalk Server en BAM': http://download.microsoft.com/download/7/9/8/79802a44-7903-402c-a1a7-e7442aa29382/Net8_p64_67_118.pdf

Rémon ter Haar (remon.ter.haar@logica.com), is als Experienced Software Engineer werkzaam bij Logica en bij verschillende BizTalk projecten betrokken geweest.



Vincent Scheel (vincent@it-grator.com), werkt sinds kort als freelance BizTalk consultant en houdt zich voornamelijk bezig met applicatie integratie met behulp van Microsoft BizTalk Server en .Net.

