

Thinkwise moderniseert legacy-systemen vanuit het hart

Voor veel bedrijven zijn legacy-systemen steeds meer een remmende factor op hun ontwikkeling. Bedrijven beschikken vaak over onmisbare informatiesystemen die functioneel en technologisch snel verouderen. De IT-afdelingen durven deze legacy-systemen bijna niet aan te passen omdat de functionele kennis ervoor ontbreekt of omdat het risico te groot is dat de ontwikkelaars fouten introduceren. Daarnaast hebben de systemen vaak nog tekstgebaseerde schermen en kan er geen gebruikgemaakt worden van faciliteiten zoals webservices. Nog een probleem is dat de arbeidsmarkt over steeds minder ontwikkelaars beschikt met voldoende kennis van de oude technieken, want studenten op hogescholen en universiteiten leren al jaren geen RPG of Cobol meer. Dit alles heeft tot gevolg dat informatiesystemen niet meer mee veranderen met de organisatie. En dat terwijl snel inspelen op een veranderende wereld belangrijker is dan ooit.

Huidige alternatieven: herbouw, code conversie en SOA

De problemen die legacy-systemen opleveren worden steeds groter waardoor bedrijven het moderniseren niet langer voor zich uit kunnen schuiven. De grote vraag is natuurlijk: hoe pak je zo'n moderniseringsslag aan? Ten eerste kun je er voor kiezen om het gehele systeem opnieuw te bouwen. Dit levert volledige vrijheid op om technische en functionele keuzes te maken, maar traditionele problemen van maatwerktrajecten komen direct om de hoek kijken. Denk hierbij aan lange doorlooptijden, veranderende specificaties en daardoor moeilijk in te plannen werkzaamheden. Een andere methode is het gebruik van code convertors. Deze convertors vertalen bijvoorbeeld RPG-code naar moderne talen als C# en Java. Dit scheelt in eerste instantie veel werk, maar meestal is het eindresultaat erg onoverzichtelijk en neemt de hoeveelheid code sterk toe. En hiermee is het werk nog niet gedaan: de ontwikkelaar moet nog heel veel code met de hand aanpassen of toevoegen voordat het systeem daadwerkelijk operationeel is.

Verlengde levensduur

Het gebruik van SOA is een derde manier om de levensduur van legacy-systemen te verlengen. De ontwikkelaar verpakt de RPG-functionaliteit in een schil van web-



services. Het voordeel hiervan is dat de huidige functionaliteit operationeel blijft, terwijl de gebruiker een veel modernere applicatie krijgt te zien en koppelingen naar andere systemen veel eenvoudiger zijn. Het probleem van deze aanpak is de toename van overhead en dat de ontwikkelaar nog steeds RPG-code moet aanpassen zodra er functionele wijzigingen nodig zijn. Dit is slechts uitstel van executie.

Alle genoemde methoden helpen bij het moderniseren van legacy-systemen, maar geven geen antwoord op de volgende vragen:

- Hoe moderniseer ik mijn volledige legacy-systeem zonder in een onbeheersbaar ontwikkelproject terecht te komen?
- Hoe voorkom ik dat mijn gemoderniseerde systeem over tien jaar weer een legacy-systeem is?

De sleutel tot succes

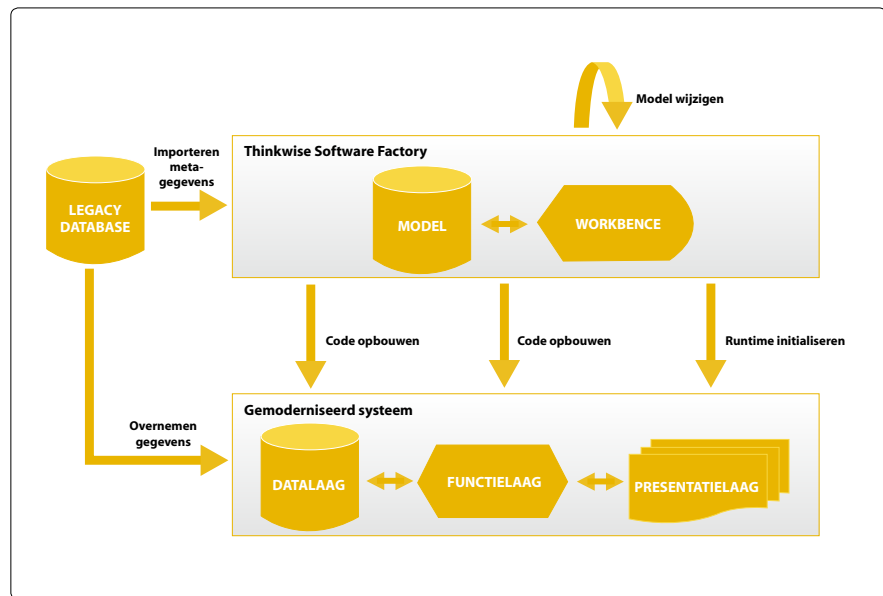
De oplossing voor duurzame modernisering ligt in de scheiding tussen technologie en functionaliteit. Zodra je in staat bent om de technologische en functionele ontwikkelingen los van elkaar door te voeren, kun je technologische veranderingen inpassen terwijl de functionaliteit intact blijft. Thinkwise Software levert een Software Factory die deze scheiding realiseert.

De Thinkwise Software Factory is een ontwikkelgereedschap voor het bouwen van een informatiesysteem dat bestaat uit een presentatielaag, een functielaag en een data-laag. De bron van het systeem is het model van het eindproduct dat wordt opgeslagen in de Software Factory database. Dit model bevat niet alleen gegevens over het datamodel, maar beschrijft ook de functielaag en de presentatielaag. Grafische tools binnen de Workbench

ondersteunen de ontwikkelaar bij het opzetten van het model. De ontwikkelaar voert functionele wijzigingen altijd door vanuit het model en maakt hierbij optimaal gebruik van methoden als Aspect Oriented Programming en Agile Programming. Vanuit het model bouwt de Thinkwise Software Factory code voor de data laag en de functielaag. De presentatielaag bestaat uit een applicatie die zichzelf bij opstarten configureert op basis van het model.

Moderniseren vanuit het hart

Hoe moderniseert de Thinkwise Software Factory legacy-systemen? Het datamodel vormt de basis van het legacy-systeem. Het hart van het nieuwe informatiesysteem is het model in de Software Factory. De kunst is om dit model te vullen aan de hand van informatie over en uit het legacy-systeem. Daarvoor wordt gebruik gemaakt van onder andere de systeemtabellen waaruit het (data)model grotendeels kan worden afgeleid. Vervolgens kan de ontwikkelaar het model aanpassen en uitbreiden om daarmee de data laag te optimaliseren. Zodra dit gereed is, bouwt de Thinkwise Software Factory de data laag en een architectuur voor de functielaag, waarna een importfunctie de gegevens kopieert vanuit het legacy-systeem naar de data laag. Hierna richt de ontwikkelaar de presentatielaag naar eigen smaak in met de Workbench. Het resultaat is een informatiesysteem met dezelfde gegevens als het legacy-systeem en een volledige operationele presentatielaag in C# (.NET), J2SE en J2EE. Dit resultaat is binnen enkele dagen te bereiken en is daarom zeer geschikt voor een proefproject. De functielaag bestaat uit functionaliteit die de bedrijfslogica bevat zoals controles en berekeningen. De ontwerper vult het model aan met beschrijvingen van deze functionaliteit op basis van de specificaties van het legacy-systeem. Vervolgens weeft de ontwikkelaar broncode templates in de functielaag op basis van die beschrijvingen met moderne talen als C#, Java en SQL. Ter verduidelijking van dit concept kan de ontwikkelaar in een proefproject een deel van de functionaliteit bouwen.



Voordelen

Moderniseren van legacy-systemen met de Thinkwise Software Factory biedt een aantal voordelen:

- De presentatielaag is direct beschikbaar in diverse technologieën zowel onder Web als Windows. Functionaliteit als sorteren, zoeken, filteren, afdrukken en Office integratie zijn beschikbaar zonder handmatig programmeerwerk. Zodra Thinkwise een nieuwe technologie in de presentatielaag beschikbaar stelt, kan de ontwikkelaar deze direct inzetten bij de eindgebruiker.
- De Software Factory maakt geen gebruik van een specifieke Thinkwise taal. De functielaag bestaat uit talen als bijvoorbeeld Java, C# en SQL, waarbij minder gecodeerd wordt door het weven van de codetemplates. RPG- en/of COBOL experts kiezen hieruit zelf een programmeertaal die goed bij hen past.
- Nadat het systeem is gemoderniseerd, kan het eenvoudig worden doorontwikkeld. Een wijziging in het model werkt direct door in alle lagen. Hierdoor kan de ontwikkelaar wijzigingen in het systeem zonder risico doorvoeren. Zodra de ontwikkelaar bijvoorbeeld een veld aan een tabel toevoegt, reageren de data laag, de functielaag en de

presentatielaag hier direct op met als gevolg dat interfaceverschillen tussen de lagen niet meer bestaan.

Duurzame modernisering

De Thinkwise Software Factory geeft duidelijke antwoorden op de belangrijkste vraagstukken van het moderniseren van legacy-systemen. De modelgedreven softwarebouw maakt ontwikkeltrajecten flexibel en beheersbaar waardoor bedrijven met een gerust hart hun maatwerksysteem kunnen moderniseren. De scheiding tussen technologie en functionaliteit biedt duurzaamheid van het informatiesysteem, zodat bedrijven over tien jaar niet weer met een legacy-systeem zitten. De Thinkwise Software Factory zorgt hiermee voor de belangrijkste randvoorwaarden voor een succesvol moderniseringstraject en biedt organisaties kansen om duurzaam te innoveren. ■



Boogschutterstraat 7b,
7324 AE Apeldoorn
t. 055-3128280
e. info@thinkwisesoftware.com
i. www.thinkwisesoftware.com