

## Mit Verastream Host Integrator neue Wege in der Integration beschreiten

### Abstraktion auf hoher Ebene

#### CONTENTS

Mainframeanwendungen: Die (schwierige) Natur der Sache .....	1
Beispiel zur Anwendungsprogrammierung .....	2
Weniger Netzwerkverkehr: Ein zusätzlicher Vorteil .....	5
Schnelle Hostintegration: Das Maß aller Dinge .....	5
Umfangreiche Integrationslösungen .....	5
Attachmate .....	5

# Mit Verastream Host Integrator neue Wege in der Integration beschreiten

## Abstraktion auf hoher Ebene

Zeit ist Geld - beides ist bei den meisten IT-Organisationen heutzutage knapp. Doch die zunehmende Zahl der Kundenanwendungen und die damit verbundenen hohen Erwartungen technisch versierter Kunden lassen den Unternehmen keine Wahl. Sie sind gezwungen, Geschäftspartnern und Kunden Zugang zu anspruchsvollen Services zu eröffnen. Und das alles mit dem vorhandenen Know-how bei einer immer dünner werdenden Personaldecke.

Auch aus technischer Sicht sind die Anforderungen erheblich. Die Mehrzahl der Programme und Daten zur Unterstützung neuer Kundenanwendungen befindet sich immer noch auf Mainframesystemen. Da in den meisten Mainframeanwendungen nie eine klare Trennung von Programmen und Daten vollzogen worden ist, ist die Integration dieser Anwendungen seit jeher schwierig und zeitaufwendig.

Attachmate® Verastream® Host Integrator eröffnet einen völlig neuen Weg zur Integration von Mainframeanwendungen. Die unstrukturierten Programme und Daten werden hierbei in Dienste umgesetzt, damit Anwendungsentwickler strukturiert darauf zugreifen können. Der vorliegende Fachbeitrag stellt das High-Level-Abstraktionskonzept von Verastream Host Integrator aus technischer Sicht vor und erläutert die Verwendung von Tabellen und Prozeduren zur Beschleunigung des Integrationsprozesses.

### Mainframeanwendungen: Die (schwierige) Natur der Sache

Der programmtechnische Zugriff auf Programme und Daten einer Mainframeanwendung über die Terminalschnittstelle beginnt üblicherweise mit einem Modellierungsprozess. Im Laufe dieses Prozesses werden bestimmte Bildschirme ausgewählt, Dateneingaben und -ausgaben definiert und Benutzerinteraktionen aufgezeichnet. Die ursprüngliche Mainframefunktionalität wird als Ergebnis dieses Prozesses in einem Service gekapselt, der auf einem zwischengeschalteten Server eingesetzt werden kann. Der Zugriff darauf erfolgt über eine API.

Die Handhabung von Mainframeanwendungen über eine API kann überaus komplex sein. Wie in Abb. 1 gezeigt, kann eine typische Mainframeanwendung mehrere Eingabefelder, eine Menge von Datensätzen, die sich über mehrere Bildschirme erstrecken, und verschiedene Navigationsoptionen umfassen. Um eine Hosttransaktion abschließen zu können, ist zudem die Interaktion mit mehreren Anwendungsbildschirmen erforderlich.

Die meisten Mainframe-Integrationsprodukte stellen APIs zur Adressierung der Anwendung auf einer relativ niedrigen Abstraktionsebene bereit. Anwendungsentwickler benötigen daher ein detailliertes Verständnis der Mainframeanwendung, um zwischen den Bildschirmen der Anwendung zu navigieren, Daten einzugeben und Daten zu extrahieren. Abb. 1 zeigt den typischen Bildschirm einer Hostanwendung mit zugehörigen Eingabefeldern, Datensätzen und Navigationsbefehlen.

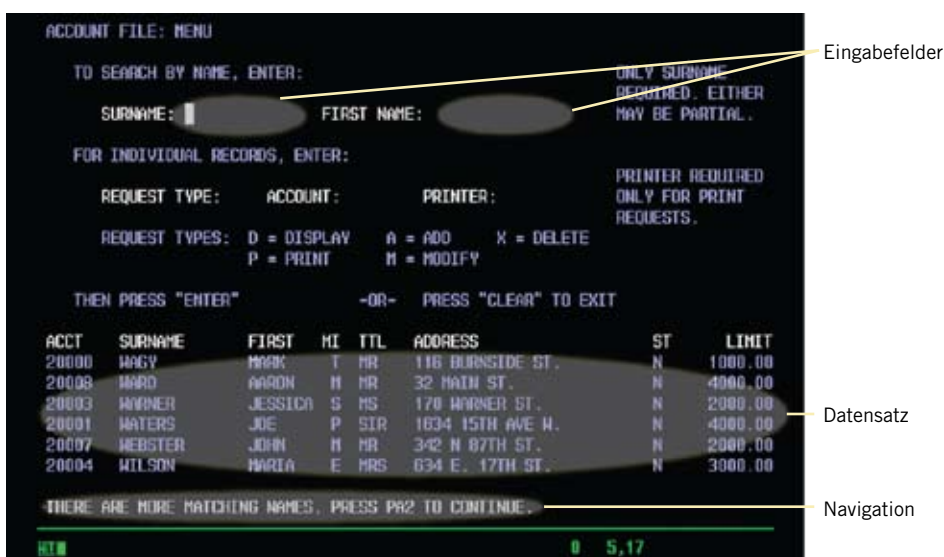


Abb. 1: Bildschirm einer Hostanwendung

Im Unterschied zu anderen Konzepten kombiniert Verastream Host Integrator eine funktionsstarke Sammlung von Java- .NET- und Web-Service-APIs mit höheren Abstraktionsebenen, wodurch es Anwendungsentwicklern erspart bleibt, sich mit der Komplexität der zugrundeliegenden Hostanwendungen im Detail auseinanderzusetzen. Abb. 2 zeigt die Abstraktionshierarchie in einem von Verastream Host Integrator erzeugten Service. Die APIs in Verastream Host Integrator ermöglichen den Zugang auf jeder Abstraktionsebene.

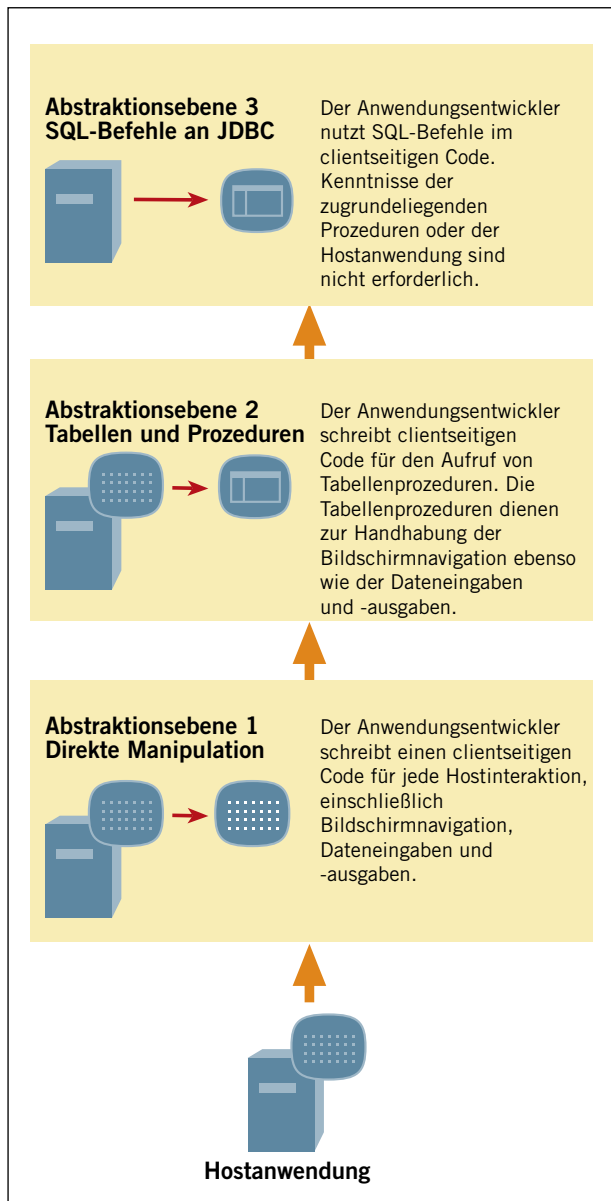


Abb. 2: Abstraktionsebenen von Verastream Host Integrator

Während des Modellierungsprozesses mit Verastream definieren Fachleute, die mit der Hostanwendung vertraut sind, Tabellen mit Datenbankspalten, in denen die jeweiligen Datenelemente benannt sind. Prozeduren ordnen Dateneingaben und -ausgaben aus beliebigen Bildschirmen der Hostanwendung zu und übernehmen automatisch die Handhabung aller Details der Navigations- und Transaktionsabläufe. Prozeduren - einschließlich Auswählen, Aktualisieren, Einfügen und Löschen - können miteinander kombiniert werden, um eine komplexe Transaktionsreihe durchzuführen.

Anhand der mit Verastream Host Integrator erstellten Tabellen und Prozeduren können Anwendungsentwickler mit unstrukturierten Hostanwendungen genauso arbeiten wie mit einer beliebigen Standarddatenbank. Abb. 3 zeigt den Dialog anhand von Tabellen und Prozeduren aus Verastream Host Integrator.

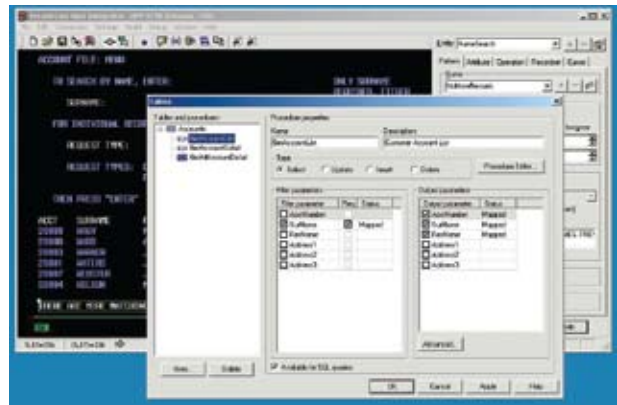


Abb. 3: Tabellen und Prozeduren von Verastream Host Integrator

### Beispiel zur Anwendungsprogrammierung

Welchen Wert die Abstraktion auf hoher Ebene für die Vereinfachung der Integration hat, lässt sich an einem einfachen Programmierbeispiel darstellen. In diesem Beispiel möchte ein Benutzer einer Hostanwendung Informationen über alle Kunden abrufen, deren Nachnamen mit „w“ beginnen.

Bei einer nicht integrierten Mainframeanwendung beginnt diese Transaktion damit, dass der Benutzer den Buchstaben „w“ eingibt. Die Anwendung gibt eine Kontenliste aller Kunden aus, deren Nachnamen mit „w“ beginnen. An dieser Stelle muss der Benutzer jede Kontennummer per Hand eingeben, um die gewünschten Informationen einzeln abzurufen. Dieser Vorgang verlangt die Durchführung wiederholter Arbeitsschritte in einer Reihe von Anwendungsbildschirmen. Es ist nicht möglich, alle Informationen in einem einzigen Bildschirm zusammenzufassen oder die Ergebnisse nach Kontennummern zu sortieren.

Mit einer neu integrierten Geschäftsanwendung kann der Benutzer durch bloße Eingabe des Buchstabens „w“ deutlich bessere Ergebnisse erzielen. Mit einer einzigen Operation könnte er beispielsweise die in Abb. 4 gezeigte Tabelle erzeugen.

<b>Account Number</b>	<b>Last Name</b>	<b>First Name</b>	<b>Address 1</b>	<b>Address 2</b>
20000	Wagy	Mark	116 Burnside St	Portland, OR 97219
20001	Waters	Joe	1634 15th Ave. W	San Diego, CA 92101
20003	Warner	Jessica	170 Warner St	Warner, MN 26582
20004	Wilson	Maria	634 E 17th St	Taos, NM 98765
20006	Winters	Shelly	1400 Snowy Lane	Portland, OR 38694
20007	Webster	John	342 N 87th St	Seattle, WA 98456
20008	Ward	Aaron	32 Main St	Plymouth, IN 74856

Abb. 4: Ausgabe einer Anwendungstransaktion

Es gibt zwei Möglichkeiten, die Tabelle in Abb. 4 zu erzeugen. Entweder mit APIs auf niedriger oder auf hoher Abstraktionsebene. Abb. 5 vergleicht für jedes Verfahren die hinter den Kulissen notwendigen Arbeiten. Die mittlere Spalte zeigt, welcher Programmieraufwand bei Verwendung von APIs auf niedriger Abstraktionsebene nötig ist, um die Mainframeanwendung direkt anzusprechen. Im Unterschied dazu zeigt die rechte Spalte, wie das gleiche Ergebnis mit einem einzigen SQL-Befehl erzielt werden kann. Grundlage dafür sind die aus Verastream Host Integrator erstellten Tabellen und Prozeduren.

Schritte	Anwendungscode mit API auf niedriger Abstraktionsebene	Anwendungscode mit Tabellen und Prozeduren aus Verastream
<p>Buchstabe „w“ in die Hostanwendung eingeben Zum Bildschirm mit den Suchergebnissen navigieren</p> <p>Die ausgegebenen Kontonummern abrufen</p> <p>Die Liste der Kontonummern sortieren</p>	<pre> HashMap attributesMap = new HashMap(); Vector fieldNames = new Vector(); Vector detailRecords = new Vector(); attributesMap.put("LastName", "w"); // Buchstabe w an Attribut LastName uebergeben mySession.setAttributes(attributesMap); // zum Element NameSearchResults navigieren // Zum Hauptmenü zurückkehren mySession.setCurrentEntity ("NameSearchResults"); fieldNames.add("AcctNum"); // Datensätze mit Feldnamen AcctNum holen records = mySession.fetchRecords (0, fieldNames, null); // Array von Kontonummern erstellen String[] acctArray = new String [records.getCount()]; for (int i = 0; i &lt; records.getCount(); i++) {     acctArray[i] = records.get(i).get("AcctNum"); } // Array von Kontonummern sortieren Arrays.sort(acctArray); // zusätzliche Attribute für jeden Eintrag im Datensatz fieldNames.add("LastName"); fieldNames.add("FirstName"); fieldNames.add("Address1"); fieldNames.add("Address2"); // Array von Kontonummern aus der Namensuche schrittweise durchlaufen, um die Kontendetails abzurufen for (int i = 0; i &lt; acctArray.length; i++) {     // zum Element Main navigieren     mySession.setCurrentEntity("Main"); </pre>	
<p>Eine Kontonummer aus der Liste eingeben</p> <p>Eine Kontonummer an die Hostanwendung übergeben</p> <p>Zum Bildschirm mit dem Datensatz navigieren</p> <p>Die entsprechenden Kontenangaben abrufen</p> <p>Die Kontenangaben für eine zukünftige Verwendung speichern Zum Hauptmenü zurückkehren Schritte für alle Kontenangaben wiederholen</p>	<pre> // Kontonummer aus dem aktuellen Datensatz holen und an Attribut AcctNum uebergeben attributesMap.put("AcctNum", acctArray[i]); // Attribut AcctNum schreiben  mySession.setAttributes(attributesMap);  // zum Element RecordDisplay navigieren mySession.setCurrentEntity("RecordDisplay");  // Attribute für jede Kontonummer einlesen record = mySession.getAttributes(fieldNames);  // Kontodetails in einem Array aus Datensätzen speichern detailRecords.add(record); } </pre>	<pre> AppConnRecordSet records = mySession.executeSQLStatement("Select AcctNum, LastName, FirstName, Address1, Address2 from Accounts where LastName like 'w' order by AcctNum"); </pre>

Abb. 5: Codierung im Vergleich

Das hier gezeigte Transaktionsbeispiel ist recht einfach gehalten. Mit zunehmender Komplexität der Transaktionen nehmen die Vorteile durch eine beschleunigte Entwicklung unter Verwendung der Tabellen und Prozeduren aus Verastream drastisch zu.

### Weniger Netzwerkverkehr: Ein zusätzlicher Vorteil

Die schnelle Anwendungsentwicklung ist nur ein Teil der Beschleunigung, die durch den Einsatz der Tabellen und Prozeduren aus Verastream möglich wird.

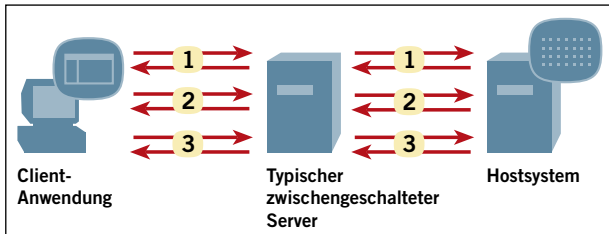


Abb. 6: Von Low-Level-DPIs erzeugter Verkehr im Netzwerk

Werden Client-Anwendungen so geschrieben, dass sie direkt mit Mainframeanwendungen interagieren, muss jede Interaktion das Netzwerk einmal vollständig durchlaufen (siehe Abb. 6). Der in Abb. 5 gezeigte API-Code auf niedriger Abstraktionsebene erzeugt z. B. auf diese Weise mehr als 20 Durchläufe im Netz.

Werden Client-Anwendungen so geschrieben, dass sie die Tabellen und Prozeduren aus Verastream nutzen, reduziert sich der Verkehr im Netzwerk, und die Transaktionsverarbeitung wird von der Client-Anwendung auf den zwischengeschalteten Verastream-Server verlagert.

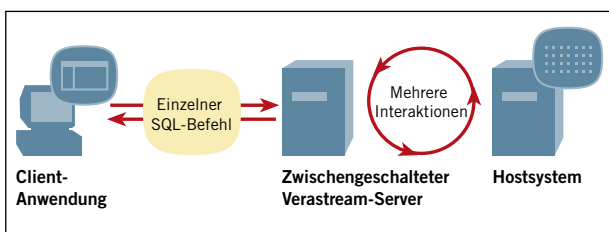


Abb. 7: Von einem einzelnen SQL-Befehl erzeugter Verkehr im Netzwerk

Abb. 7 zeigt, dass ein einzelner SQL-Befehl, der von dem Anwendungscode anhand der Tabellen und Prozeduren erzeugt wird, einen einzigen Durchlauf im Netzwerk zwischen Client-Anwendung und Server auslöst, auf dem Verastream Host Integrator ausgeführt wird. Der zur Transaktionsverarbeitung

erforderliche Dialog erfolgt nur zwischen dem Server und der Mainframe-Anwendung, wodurch sich der gesamte Verkehr im Netz deutlich verringert, was wiederum zu einer höheren Gesamtleistung beiträgt.

### Schnelle Hostintegration: Das Maß aller Dinge

Die Innovationsgeschwindigkeit nimmt zu, während die IT-Abteilungen gleichzeitig gezwungen sind, mehr Arbeit mit weniger Ressourcen zu bewältigen. Die von Verastream Host Integrator erzeugte einzigartige Darstellung unstrukturierter Mainframe-Anwendungen in Form von Tabellen und Prozeduren beschleunigt die Integration, weil die Komplexität der Entwicklungsaufgaben sinkt und die vorhandenen Ressourcen und Fähigkeiten optimal genutzt werden können. Auch im laufenden Betrieb trägt Verastream Host Integrator zu einer Beschleunigung der Verarbeitung über eine Entlastung des Datenaufkommens im Netzwerk bei.

### Umfangreiche Integrationslösungen

Verastream Host Integrator gehört zur Attachmate Verastream Produktlinie, eine Familie aus Werkzeugen zur Integration von Mainframe- und Enterprise-Anwendungen in serviceorientierte Architekturen. Da bestehende Anwendungen nicht umgeschrieben werden müssen, entfällt das damit verbundene Risiko. Die Integration der Anwendungen und die Anwendungsentwicklung werden gleichzeitig beschleunigt und die Arbeitsabläufe werden verbessert.

Verastream Host Integrator macht sämtliche Hostanwendungen für die Integration in SOA-Umgebungen zugänglich, indem Geschäftsprozesse als Web-Services, XML-, Java- oder .NET-Komponenten zur Verfügung gestellt werden. Weil vorhandene Entwicklungsressourcen und vertraute IT-Tools eingesetzt werden können, ist die gewünschte Integration zudem deutlich schneller zu erreichen. Unabhängig davon, ob in der Hostumgebung IBM System z (S/390), IBM System i (AS/400), UNIX, OpenVMS oder HP e3000 eingesetzt werden, Verastream Host Integrator trägt zur Modernisierung der Legacy-Systeme bei, ohne die täglichen Geschäftsabläufe zu beeinträchtigen.

### Attachmate

Attachmate ist ein führender Hersteller von Softwarelösungen für die Terminalemulation, Applikationsintegration und sichere Kommunikation. Der Geschäftsbereich NetIQ bietet innovative Lösungen zur Automatisierung von IT-Prozessen und zur Gewährleistung von Leistung, Sicherheit und Compliance in verteilten IT-Umgebungen. Weltweit setzen mehr als 65.000 Anwender auf unsere Technologien, um ihre IT-Ressourcen optimal zu nutzen. [www.attachmate.de](http://www.attachmate.de).



**Hauptsitz**  
1500 Dexter Avenue North  
Seattle, Washington 98109  
TEL +1 206 217 7500  
FAX +1 206 217 7515

**Europäische Zentrale**  
Niederlande  
TEL +31 172 50 55 55  
FAX +31 172 50 55 51

**Österreich**  
TEL +43 1 595 4335 0  
FAX +43 1 595 4335 11  
[www.attachmate.at](http://www.attachmate.at)  
info-at@attachmate.com

**Schweiz**  
TEL +41 43 399 2090  
FAX +41 43 399 2099  
[www.attachmate.ch](http://www.attachmate.ch)  
InfoCH@attachmate.com

**Deutschland**  
TEL +49 89 99 351 0  
FAX +49 89 99 351 111  
TEL +49 2102 4965 0  
FAX +49 2102 4965 65  
[www.attachmate.de](http://www.attachmate.de)  
info-de@attachmate.com

Weitere Informationen finden Sie unter [www.attachmate.de](http://www.attachmate.de)